Studies in Computational Intelligence 1190

Hocine Cherifi · Murat Donduran · Luis M. Rocha · Chantal Cherifi · Onur Varol *Editors*

Complex Networks & Their Applications XIII

Proceedings of The Thirteenth International Conference on Complex Networks and their Applications: COMPLEX NETWORKS 2024 - Volume 4

Springer

Studies in Computational Intelligence 1190

Series Editor

Janusz Kacprzyk, *Polish Academy of Sciences, Warsaw, Poland*

Editorial Board Members

Marco Dorigo, *Université Libre de Bruxelles, Bruxelles, Belgium*
Andries Engelbrecht, *University of Stellenbosch, Stellenbosch, South Africa*
Vladik Kreinovich, *University of Texas at El Paso, El Paso, TX, USA*
Francesco Carlo Morabito, *Mediterranea University of Reggio Calabria, Reggio Calabria, Italy*
Roman Slowinski, *Poznan University of Technology, Poznan, Poland*
Yingxu Wang, *Schulich School of Engineering, Calgary, AB, Canada*
Yaochu Jin, *Westlake University, Hangzhou, China*

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

Indexed by SCOPUS, DBLP, WTI AG (Switzerland), zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

Hocine Cherifi · Murat Donduran ·
Luis M. Rocha · Chantal Cherifi · Onur Varol
Editors

Complex Networks & Their Applications XIII

Proceedings of The Thirteenth International Conference on Complex Networks and their Applications: COMPLEX NETWORKS 2024 - Volume 4

Springer

Editors
Hocine Cherifi
ICB
University of Burgundy
Dijon, France

Murat Donduran
Department of Economics
Yildiz Technical University
Istanbul, Türkiye

Luis M. Rocha
Thomas J. Watson College of Engineering
and Applied Science
Binghamton University
Binghamton, USA

Chantal Cherifi
IUT Lumière
Université Lyon 2
Bron, France

Onur Varol
Faculty of Engineering and Natural Sciences
Office
Sabanci University
Istanbul, Türkiye

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
ISBN 978-3-031-82438-8 ISBN 978-3-031-82439-5 (eBook)
https://doi.org/10.1007/978-3-031-82439-5

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2025

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

Preface

Dear Colleagues, Participants, and Readers,

We are thrilled to present the proceedings of the 13th International Conference on Complex Networks and Their Applications, held in the vibrant city of Istanbul, Türkiye, from December 10 to 12, 2024. This year's conference marks yet another milestone in the growth of our interdisciplinary community, reflecting the ever-expanding significance of complex network research. Understanding the structures and dynamics that govern complex systems has never been more critical as our world becomes more interconnected.

The Complex Networks Conference has evolved over the past thirteen years into a crucial platform where scholars, researchers, and professionals from various disciplines converge. Through these proceedings, we showcase groundbreaking work that explores the diverse applications of network science, from biological systems to social networks, technological infrastructures, and economic systems. Each edition brings new ideas and breakthroughs, and this year is no exception, with significant contributions from a global cohort of experts.

We are particularly honored to feature keynote lectures from internationally renowned scholars, whose talks highlight some of the most pressing issues and exciting developments in the field:

- Federico Battiston (Central European University, Austria) presents "Higher-order network science".
- Tina Eliassi-Rad (Northeastern University, USA) explored the "Stability of Graph Machine Learning Models".
- Frank Emmert-Streib (Tampere University, Finland) discussed "From Digital Twins to Complexity Data Science".
- Filippo Menczer (Indiana University, USA) talked about "AI and Social Media Manipulation: The Good, the Bad, and the Ugly".
- Luciano Pietronero (University of Rome "Sapienza," Italy) provided deep insights into "Economic Fitness: Concepts, Methods, and Applications".

These keynote talks reflect the richness and diversity of this year's conference, touching on cutting-edge issues in both theory and application. In addition, we are excited to have offered two pre-conference tutorials on December 9, 2024:

- Alessandro Galeazzi (University of Padova, Italy) led a tutorial on "Unveiling Echo Chambers and Users' Opinions on Online Social Media".
- Clara Stegehuis (University of Twente, The Netherlands) presented a session on "Mathematical Analysis and Applications of Network Motifs".

The proceedings of COMPLEX NETWORKS 2024, published by Springer, consist of four volumes that capture a wide range of topics, including Machine Learning Networks, Network Embedding, Link Analysis Ranking, Social Networks, Biological Networks, Higher-Order Interactions, and Community Structure, among others. Each

paper included here has undergone a rigorous peer-review process, ensuring the high quality and scientific relevance that our community expects.

We are also pleased to announce that extended versions of selected papers will be invited for publication in prestigious journals, including PLOS Complex Systems, Applied Network Science, Social Network Analysis and Mining, and Advances in Complex Systems, furthering the reach and impact of the research presented at this conference.

We would like to extend our sincere gratitude to the authors, presenters, and reviewers for their contributions, which are at the core of this conference's success. Our most profound appreciation goes to the organizing committees and volunteers, whose dedication has made this year's edition successful.

We invite you to explore the rich content within these proceedings and hope that they will inspire future research, foster innovation, and stimulate collaborations within the complex networks' community. As you explore these contributions, we are confident you will share the enthusiasm and excitement we felt in organizing COMPLEX NETWORKS 2024.

Murat Donduran
Hocine Cherifi
Luis M. Rocha
Chantal Cherifi
Onur Varol

Organization and Committees

General Chairs

Hocine Cherifi	University of Burgundy, France
Luis M. Rocha	Binghamton University, USA

Advisory Board

Jon Crowcroft	University of Cambridge, UK
Raissa D'Souza	University of California, Davis, USA
Eugene Stanley	Boston University, USA
Ben Y. Zhao	University of Chicago, USA

Program Chairs

Chantal Cherifi	University of Lyon, France
Murat Donduran	Yildiz Technical University, Turkey

Lightning Chairs

Konstantin Avrachenkov	Inria Université Côte d'Azur, France
Mathieu Desroches	Inria Université Côte d'Azur, France
Huijuan Wang	TU Delft, Netherlands

Poster Chairs

Christophe Crespelle	Université Côte d'Azur, France
Manuel Marques Pita	Universidade Lusófona, Portugal
Laura Ricci	University of Pisa, Italy

Special Issues Chair

Sabrina Gaito	University of Milan, Italy

Publicity Chairs

Fabian Braesemann	University of Oxford, UK
Zachary Neal	Michigan State University, USA
Xiangjie Kong	Dalian University of Technology, China

Tutorial Chairs

Luca Maria Aiello	Nokia Bell Labs, UK
Leto Peel	Maastricht University, Netherlands

Social Media Chair

Brennan Klein	Northeastern University, USA

Sponsor Chairs

Roberto Interdonato	CIRAD-UMR TETIS, France
Christophe Cruz	University of Burgundy, France

Sustainability Chair

Madeleine Aurelle	City School International De Ferney-Voltaire, France

Local Committee Chair

Charlie Joyez	Université Côte d'Azur, France

Publication Chair

Matteo Zignani	University of Milan, Italy

Submission Chair

Cheick Ba	Queen Mary University of London, UK

Web Chairs

Stephany Rajeh	Sorbonne University, France
Alessia Galdeman	University of Milan, Italy

Program Committee

Jacobo Aguirre	Centro de Astrobiología (CAB), Spain
Luca Maria Aiello	ITU Copenhagen, Denmark
Esra Akbas	Georgia State University, USA
Sinan G. Aksoy	Pacific Northwest National Laboratory, USA
Mehmet Aktas	Georgia State University, USA
Tatsuya Akutsu	Kyoto University, Japan
Reka Albert	Pennsylvania State University, USA
Alberto Aleta	University of Zaragoza, Spain
Claudio Altafini	Linkoping University, Sweden
Viviana Amati	University of Milano-Bicocca, Unknown
Frederic Amblard	Université Toulouse 1 Capitole, IRIT, France
Enrico Amico	EPFL, Switzerland
Yuri Antonacci	University of Palermo, Italy
Alberto Antonioni	Carlos III University of Madrid, Spain
Nino Antulov-Fantulin	ETH Zurich, Switzerland
Mehrnaz Anvari	Fraunhofer SCAI, Germany
David Aparicio	Zendesk, Portugal
Nuno Araujo	Univ. de Lisboa, Portugal
Panos Argyrakis	Aristotle University of Thessaloniki, Greece
Oriol Artime	University of Barcelona, Spain
Malbor Asllani	Florida State University, USA
Tomaso Aste	University College London, UK
Martin Atzmueller	Osnabrück University & DFKI, Germany
Konstantin Avrachenkov	Inria Sophia-Antipolis, France

Giacomo Baggio	University of Padova, Italy
Franco Bagnoli	Università di Firenze, Italy
James Bagrow	University of Vermont, USA
Yiguang Bai	Xidian University, China
Sven Banisch	Karlsruhe Institute of Technology, Germany
Annalisa Barla	Università degli Studi di Genova, Italy
Nikita Basov	The University of Manchester, UK
Anais Baudot	CNRS, AMU, France
Gareth J. Baxter	University of Aveiro, Portugal
Loredana Bellantuono	University of Bari Aldo Moro, Italy
Andras Benczur	SZTAKI, Hungary
Rosa M. Benito	Universidad Politécnica de Madrid, Spain
Ginestra Bianconi	Queen Mary University of London, UK
Ofer Biham	The Hebrew University, Israel
Romain Billot	IMT Atlantique, France
Livio Bioglio	University of Turin, Italy
Hanjo D. Boekhout	Leiden University, Netherlands
Anthony Bonato	Toronto Metropolitan University, Canada
Anton Borg	Blekinge Institute of Technology, Sweden
Cecile Bothorel	IMT Atlantique, France
Federico Botta	University of Exeter, UK
Romain Bourqui	University of Bordeaux, France
Alexandre Bovet	University of Zurich, Switzerland
Dan Braha	New England Complex Systems Institute, USA
Ulrik Brandes	ETH Zürich, Switzerland
Rion Brattig Correia	Instituto Gulbenkian de Ciência, Portugal
Chico Camargo	University of Exeter, UK
Gian Maria Campedelli	Fondazione Bruno Kessler, Italy
M. Abdullah Canbaz	University at Albany SUNY, USA
Vincenza Carchiolo	DIEEI, Italy
Dino Carpentras	ETH Zürich, Switzerland
Giona Casiraghi	ETH Zürich, Switzerland
Douglas Castilho	Federal Inst. of South of Minas Gerais, Brazil
Costanza Catalano	University of Florence, Italy
Lucia Cavallaro	Free University of Bozen/Bolzano, Italy
Remy Cazabet	University of Lyon, France
Jianrui Chen	Shaanxi Normal University, China
Po-An Chen	National Yang Ming Chiao Tung Univ., Taiwan
Xihui Chen	University of Luxembourg, Luxembourg
Sang Chin	Boston University, USA
Daniela Cialfi	Institute for Complex Systems, Italy
Giulio Cimini	University of Rome Tor Vergata, Italy

Matteo Cinelli	Sapienza University of Rome, Italy
Salvatore Citraro	University of Pisa, Italy
Jonathan Clarke	Imperial College London, UK
Richard Clegg	QMUL, UK
Reuven Cohen	Bar-Ilan University, Israel
Jean-Paul Comet	Université Côte d'Azur, France
Marco Coraggio	Scuola Superiore Meridionale, Italy
Michele Coscia	ITU Copenhagen, Denmark
Christophe Crespelle	Université Côte d'Azur, France
Regino H. Criado Herrero	Universidad Rey Juan Carlos, Spain
Marcelo V. Cunha	Instituto Federal da Bahia, Brazil
David Soriano-Paños	Instituto Gulbenkian de Ciência, Portugal
Joern Davidsen	University of Calgary, Canada
Toby Davies	University of Leeds, UK
Caterina De Bacco	Max Planck Inst. for Intelligent Systems, Germany
Pietro De Lellis	University of Naples Federico II, Italy
Pasquale De Meo	University of Messina, Italy
Domenico De Stefano	University of Trieste, Italy
Fabrizio De Vico Fallani	Inria-ICM, France
Charo I. del Genio	Coventry University, UK
Robin Delabays	HES-SO, Switzerland
Yong Deng	Univ. of Electronic Science and Tech., China
Mathieu Desroches	Inria Centre at Université Côte d'Azur, France
Carl P. Dettmann	University of Bristol, UK
Zengru DI	Beijing Normal University, China
Riccardo Di Clemente	Northeastern University London, UK
Branco Di Fátima	University of Beira Interior (UBI), Portugal
Alessandro Di Stefano	Teesside University, UK
Ming Dong	Central China Normal University, China
Constantine Dovrolis	Georgia Tech, USA
Maximilien Dreveton	EPFL, Switzerland
Ahlem Drif	University of Setif, Algeria
Johan L. Dubbeldam	Delft University of Technology, Netherlands
Jordi Duch	Universitat Rovira i Virgili, Spain
Cesar Ducruet	CNRS, France
Mohammed El Hassouni	Mohammed V University in Rabat, Morocco
Frank Emmert-Streib	Tampere University, Finland
Gunes Ercal	Southern Illinois University Edwardsville, USA
Alejandro Espinosa-Rada	ETH Zürich, Switzerland
Alexandre Evsukoff	Universidade Federal do Rio de Janeiro, Brazil
Mauro Faccin	University of Bologna, Italy

Max Falkenberg	City University, UK
Guilherme Ferraz de Arruda	CENTAI Institute, Italy
Andrea Flori	Politecnico di Milano, Italy
Manuel Foerster	Bielefeld University, Germany
Emma Fraxanet Morales	Pompeu Fabra University, Spain
Angelo Furno	LICIT-ECO7, France
Sergio Gómez	Universitat Rovira i Virgili, Spain
Sabrina Gaito	Università degli Studi di Milano, Italy
José Manuel Galán	Universidad de Burgos, Spain
Alessandro Galeazzi	Ca' Foscari University of Venice, Italy
Lazaros K. Gallos	Rutgers University, USA
Joao Gama	INESC TEC-LIAAD, Portugal
Jianxi Gao	Rensselaer Polytechnic Institute, USA
David Garcia	University of Konstanz, Germany
Floriana Gargiulo	CNRS, France
Michael T. Gastner	Singapore Institute of Technology, Singapore
Alexander Gates	University of Virginia, USA
Alexandra M. Gerbasi	Exeter Business School, UK
Fakhteh Ghanbarnejad	Potsdam Inst. for Climate Impact Res., Germany
Cheol-Min Ghim	Ulsan National Inst. of Science and Tech., South Korea
Tommaso Gili	IMT School for Advanced Studies Lucca, Italy
Silvia Giordano	Univ. of Applied Sciences of Southern Switzerland, Switzerland
Rosalba Giugno	University of Verona, Italy
Kimberly Glass	Brigham and Women's Hospital, USA
David Gleich	Purdue University, USA
Antonia Godoy Lorite	UCL, UK
Kwang-Il Goh	Korea University, South Korea
Carlos Gracia	University of Zaragoza, Spain
Oscar M. Granados	Universidad Jorge Tadeo Lozano, Colombia
Michel Grossetti	CNRS, France
Guillaume Guerard	ESILV, France
Jean-Loup Guillaume	Université de la Rochelle-France, France
Furkan Gursoy	Bogazici University, Turkey
Philipp Hövel	Saarland University, Germany
Meesoon Ha	Chosun University, South Korea
Bianca H. Habermann	AMU, CNRS, IBDM UMR 7288, France
Chris Hankin	Imperial College London, UK
Yukio Hayashi	JAIST, Japan
Marina Hennig	Johannes Gutenberg University of Mainz, Germany

Takayuki Hiraoka	Aalto University, Finland
Marion Hoffman	Institute for Advanced Study in Toulouse, France
Bernie Hogan	University of Oxford, UK
Seok-Hee Hong	University of Sydney, Australia
Yujie Hu	University of Florida, USA
Flavio Iannelli	UZH, Switzerland
Yuichi Ikeda	Kyoto University, Japan
Roberto Interdonato	CIRAD, France
Antonio Iovanella	Univ. degli Studi Internazionali di Roma, Italy
Arkadiusz Jędrzejewski	CY Cergy Paris Université, France
Tao Jia	Southwest University, China
Jiaojiao Jiang	UNSW Sydney, Australia
Di Jin	University of Michigan, USA
Ivan Jokifá	Technology University of Delft, Netherlands
Charlie Joyez	GREDEG, Université Côte d'Azur, France
Bogumil Kamiński	SGH Warsaw School of Economics, Poland
Marton Karsai	Central European University, Austria
Eytan Katzav	Hebrew University of Jerusalem, Israel
Mehmet Kaya	Firat University, Turkey
Domokos Kelen	SZTAKI, Hungary
Mohammad Khansari	Sharif University of Technology, Iran
Jinseok Kim	University of Michigan, USA
Pan-Jun Kim	Hong Kong Baptist University, Hong Kong
Maksim Kitsak	TU Delft, Netherlands
Mikko Kivelä	Aalto University, Finland
Brennan Klein	Northeastern University, UK
Konstantin Klemm	IFISC (CSIC-UIB), Spain
Xiangjie Kong	Zhejiang University of Technology, China
Onerva Korhonen	University of Eastern Finland, Finland
Miklós Krész	InnoRenew CoE, Slovenia
Prosenjit Kundu	DA-IICT, Gandhinagar, Gujarat, India
Haewoon Kwak	Indiana University Bloomington, USA
Richard La	University of Maryland, USA
Josè Lages	Université de Franche-Comté, France
Renaud Lambiotte	University of Oxford, UK
Aniello Lampo	UC3M, Spain
Jennifer Larson	Vanderbilt University, USA
Paul J. Laurienti	Wake Forest, USA
Anna T. Lawniczak	University of Guelph, Canada, Canada
Deok-Sun Lee	KIAS, South Korea
Harlin Lee	Univ. of North Carolina at Chapel Hill, USA
Juergen Lerner	University of Konstanz, Germany

Lasse Leskelä	Aalto University, Finland
Petri Leskinen	Aalto University/SeCo, Finland
Inmaculada Leyva	Universidad Rey Juan Carlos, Spain
Cong Li	Fudan University, China
Longjie Li	Lanzhou University, China
Ruiqi LI	Beijing Univ. of Chemical Technology, China
Xiangtao Li	Jilin University, China
Hao Liao	Shenzhen University, China
Fabrizio Lillo	Università di Bologna, Italy
Giacomo Livan	University of Pavia, Italy
Giosue' Lo Bosco	Università di Palermo, Italy
hao long	Jiangxi Normal University, China
Juan Carlos Losada	Universidad Politécnica de Madrid, Spain
Laura Lotero	Universidad Nacional de Colombia, Colombia
Yang Lou	National Yang Ming Chiao Tung Univ., Taiwan
Meilian Lu	Beijing Univ. of Posts and Telecom., China
Maxime Lucas	CENTAI, Italy
Lorenzo Lucchini	Bocconi University, Italy
Hanbaek Lyu	UW-Madison, USA
Vince Lyzinski	University of Maryland, College Park, USA
Morten Mørup	Technical University of Denmark, Denmark
Leonardo Maccari	Ca'Foscari University of Venice, Italy
Matteo Magnani	Uppsala University, Sweden
Maria Malek	CY Cergy Paris University, France
Giuseppe Mangioni	University of Catania, Italy
Andrea Mannocci	CNR-ISTI, Italy
Rosario N. Mantegna	University of Palermo, Italy
Manuel Sebastian Mariani	University of Zurich, Switzerland
Radek Marik	CTU in Prague, Czech Republic
Daniele Marinazzo	Ghent University, Belgium
Andrea Marino	University of Florence, Italy
Malvina Marku	INSERM, CRCT, France
Antonio G. Marques	King Juan Carlos University, Spain
Christoph Martin	Hamburg University of Applied Sciences, Germany
Samuel Martin-Gutierrez	Complexity Science Hub Vienna, Austria
Cristina Masoller	Universitat Politecnica de Catalunya, Spain
Rossana Mastrandrea	IMT School for Advanced Studies, Italy
John D. Matta	Southern Illinois Univ. Edwardsville, USA
Carolina Mattsson	CENTAI Institute, Italy
Fintan McGee	Luxembourg IST, Luxembourg
Matus Medo	University of Bern, Switzerland

Ronaldo Menezes	University of Exeter, UK
Humphrey Mensah	Epsilon Data Management, LLC, USA
Anke Meyer-Baese	Florida State University, USA
Salvatore Micciche	UNIPA DiFC, Italy
Letizia Milli	University of Pisa, Italy
Marija Mitrovic	Institute of Physics Belgrade, Serbia
Andrzej Mizera	University of Warsaw, Poland
Chiara Mocenni	University of Siena, Italy
Roland Molontay	Budapest UTE, Hungary
Sifat Afroj Moon	University of Virginia, USA
Alfredo Morales	MIT, USA
Andres Moreira	UTFSM, Chile
Greg Morrison	University of Houston, USA
Igor Mozetic	Jozef Stefan Institute, Slovenia
Sarah Muldoon	State University of New York, Buffalo, USA
Tsuyoshi Murata	Tokyo Institute of Technology, Japan
Jose Nacher	Toho University, Japan
Nishit Narang	NIT Delhi, India
Filipi Nascimento Silva	Indiana University, USA
Muaz A. Niazi	National Univ. of Science & Technology, Pakistan
Peter Niemeyer	Leuphana University Lüneburg, Germany
Jordi Nin	ESADE, Universitat Ramon Llull, Spain
Rogier Noldus	Ericsson, Netherlands
Masaki Ogura	Osaka University, Japan
Andrea Omicini	Università di Bologna, Italy
Gergely Palla	Eötvös University, Hungary
Fragkiskos Papadopoulos	Cyprus University of Technology, Cyprus
Symeon Papadopoulos	Centre for Research & Technology, Greece
Alice Patania	University of Vermont, USA
Leto Peel	Maastricht University, Netherlands
Hernane B. B. Pereira	Senai Cimatec, Brazil
Josep Perelló	Universitat de Barcelona, Spain
Anthony Perez	Université d'Orléans, France
Juergen Pfeffer	Technical University of Munich, Germany
Carlo Piccardi	Politecnico di Milano, Italy
Pietro Hiram Guzzi	Univ. Magna Graecia of Catanzaro, Italy
Yoann Pigné	Université Le Havre Normandie, France
Bruno Pinaud	University of Bordeaux, France
Flavio L. Pinheiro	Universidade Nova de Lisboa, Portugal
Manuel Pita	Universidade Lusófona, Portugal
Clara Pizzuti	CNR-ICAR, Italy

Jan Platos	VSB-Technical University of Ostrava, Czech Republic
Pawel Pralat	Toronto Metropolitan University, Canada
Rafael Prieto-Curiel	Complexity Science Hub, Austria
Daniele Proverbio	University of Trento, Italy
Giulia Pullano	Georgetown University, USA
Rami Puzis	Ben-Gurion University of the Negev, Israel
Christian Quadri	Università degli Studi di Milano, Italy
Hamid R. Rabiee	Sharif University of Technology, Iran
Filippo Radicchi	Indiana University, USA
Giancarlo Ragozini	University of Naples Federico II, Italy
Juste Raimbault	IGN-ENSG, France
Sarah Rajtmajer	Penn State, USA
Gesine D. Reinert	University of Oxford, UK
Elisabeth Remy	Institut de Mathématiques de Marseille, France
Xiao-Long Ren	Univ. of Electronic Science and Tech., China
Laura Ricci	University of Pisa, Italy
Albano Rikani	INSERM, France
Luis M. Rocha	Binghamton University, USA
Luis E. C. Rocha	Ghent University, Belgium
Fernando E. Rosas	Imperial College London, UK
Giulio Rossetti	CNR-ISTI, Italy
Camille Roth	CNRS/CMB/EHESS, France
Celine Rozenblat	UNIL, Switzerland
Giancarlo Ruffo	Univ. degli Studi del Piemonte Orientale, Italy
Arnaud Sallaberry	University of Montpellier, France
Hillel Sanhedrai	Northeastern University, USA
Iraj Saniee	Bell Labs, Nokia, USA
Antonio Scala	CNR Institute for Complex Systems, Italy
Michael T. Schaub	RWTH Aachen University, Germany
Irene Sendiña-Nadal	Universidad Rey Juan Carlos, Spain
Mattia Sensi	Politecnico di Torino, Italy
Ke-ke Shang	Nanjing University, China
Julian Sienkiewicz	Warsaw University of Technology, Poland
Per Sebastian Skardal	Trinity College, Ireland
Fiona Skerman	Uppsala University, Sweden
Oskar Skibski	University of Warsaw, Poland
Keith M. Smith	University of Strathclyde, UK
Igor Smolyarenko	Brunel University, UK
Zbigniew Smoreda	Orange Innovation, France
Annalisa Socievole	ICAR-CNR, Italy
Igor M. Sokolov	Humboldt University Berlin, Germany

Albert Solé-Ribalta	Universitat Oberta de Catalunya, Spain
Sara Sottile	University of Trento, Italy
Sucheta Soundarajan	Syracuse University, USA
Jaya Sreevalsan-Nair	IIIT Bangalore, India
Christoph Stadtfeld	ETH Zürich, Switzerland
Clara Stegehuis	University of Twente, Netherlands
Lovro Šubelj	University of Ljubljana, Slovenia
Xiaoqian Sun	Beihang University, China
Michael Szell	IT University of Copenhagen, Denmark
Boleslaw Szymanski	Rensselaer Polytechnic Institute, USA
Andrea Tagarelli	University of Calabria, Italy
Kazuhiro Takemoto	Kyushu Institute of Technology, Japan
Frank W. Takes	Leiden University, Netherlands
Fabien Tarissan	CNRS & ENS Paris Saclay, France
Laura Temime	Cnam, France
François Théberge	TIMC, France
Guy Theraulaz	Université Paul Sabatier and CNRS, France
I-Hsien Ting	National University of Kaohsiung, Taiwan
Michele Tizzani	ISI Foundation, Italy
Michele Tizzoni	University of Trento, Italy
Olivier Togni	University of Burgundy, France
Leo Torres	Northeastern University, USA
Sho Tsugawa	University of Tsukuba, Japan
Francesco Tudisco	The University of Edinburgh, UK
Melvyn S. Tyloo	Los Alamos National Lab, USA
Stephen M. Uzzo	National Museum of Mathematics, USA
Lucas D. Valdez	IFIMAR-UNMdP, Argentina
Pim Van der Hoorn	Eindhoven University of Technology, Netherlands
Piet Van Mieghem	Delft University of Technology, Netherlands
Fabio Vanni	University of Insubria, Italy
Christian L. Vestergaard	Institut Pasteur, France
Tiphaine Viard	Télécom Paris, France
Julian Vicens	Eurecat, Spain
Blai Vidiella	CSIC, Spain
Pablo Villegas	Enrico Fermi Research Center (CREF), Italy
Maria Prosperina Vitale	University of Salerno, Italy
Pierpaolo Vivo	King's College London, UK
Johannes Wachs	Corvinus University of Budapest, Hungary
Huijuan Wang	Delft University of Technology, Netherlands
Lei Wang	Beihang University, China
Guanghui Wen	Southeast University, Nanjing, China
Mateusz Wilinski	Los Alamos National Laboratory, USA

Dirk Witthaut	Forschungszentrum Jülich, Germany
Bin Wu	Beijing Univ. of Posts and Telecom., China
Mincheng Wu	Zhejiang University of Technology, China
Tao Wu	Chongqing Univ. of Posts and Telecom., China
Haoxiang Xia	Dalian University of Technology, China
Gaoxi Xiao	Nanyang Technological University, Singapore
Nenggang Xie	Anhui University of Technology, China
Takahiro Yabe	MIT, USA
Kaicheng Yang	Northeastern University, USA
Yian Yin	Cornell University, USA
Jean-Gabriel Young	University of Vermont, USA
Irfan Yousuf	Univ. of Engineering and Technology, Pakistan
Yongguang Yu	Beijing Jiaotong University, China
Paolo Zeppini	University Cote d'Azur, France
Shi Zhou	University College London (UCL), UK
Wei-Xing Zhou	East China Univ. of Science and Technology, China
Eugenio Zimeo	University of Sannio, Italy
Lorenzo Zino	Politecnico di Torino, Italy
Michal R. Zochowski	University of Michigan, USA
Claudia Zucca	Tilburg University, Netherlands

Contents

Biological Networks

SimNetX: Interactive Support for Biomedical Data Analysis Using Patient Similarity Networks 3
Tomas Anlauf, Kristyna Kubikova, Eliska Ochodkova, Eva Kriegova, and Milos Kudelka

Class Dominancy Profiles in Multi-class and Multi-cluster Similarity Networks 15
Ondrej Janca, Arootin Gharibian, Eliska Ochodkova, Eva Kriegova, and Milos Kudelka

Structural Asymmetries in PPI Networks as a Tool to Improve the Detection of Protein Complexes 27
Lukas Papik, Eliska Ochodkova, Eva Kriegova, and Milos Kudelka

A Physiologically-Grounded, Computationally-Efficient Model of the Human Hippocampus 39
Niccolò Calcini, Vitalii Kyzym, and Renaud B. Jolivet

Diffusion Epidemics

Component Structure-Driven Centrality Ranking for Diffusion Management in Complex Networks 53
Issa Moussa Diop, Cherif Diallo, and Hocine Cherifi

The Impact of Heterogeneity on Epidemics: Insights from a Modified SIR Model 66
Fabio Mazza, Francesca Colaiori, Stefano Guarino, Sandro Meloni, Marco Brambilla, Carlo Piccardi, Francesco Pierri, and Fabio Saracco

The Firefighter Game with State-Varying Cost Functions 76
Ethan Hunter and Jessica Enright

Optimal Intervention Campaign to Combat Anti-vaccine Social Contagion and Contain Epidemic Spread 87
Sarah Alahmadi, Rebecca Hoyle, Michael Head, and Markus Brede

Combating Toxicity: A Systematic Approach to Model Quarantine Intervention for Varied Toxicity Levels 100
Emmanuel Addai, Nitin Agarwal, and Niloofar Yousefi

Modeling Toxicity Propagation on Reddit Using Epidemiology 113
Selimhan Dagtas, Nitin Agarwal, and Niloofar Yousefi

Dynamics on/of Networks

Classification of Sleep Stages Based on Quantile Graphs from Mouse EEG 127
Kazuki Koyama, Masanori Sakaguchi, and Takaaki Ohnishi

A Fully-Dynamic Approximation Algorithm for Maximum Weight *b*-Matchings in Graphs 139
Fabian Brandt-Tumescheit, Frieda Gerharz, and Henning Meyerhenke

Harmonizing Vs Polarizing Platform Influence Functions 152
Hind AlMahmoud and Frederik Mallmann-Trenn

Bots of a Feather: Mixing Biases in LLMs' Opinion Dynamics 166
Erica Cau, Andrea Failla, and Giulio Rossetti

Risk-Taking Influences the Network Growth and Population Dynamics in a Spatial Prisoner's Dilemma 177
Tomoko Sakiyama and Akihiro Takahara

Infrastructure Networks

Evaluating Node Contribution in Network Path Transport Using *Physarum* Centrality 187
Kiri Kawato, Tetsuya Yamamoto, and Masaki Aida

Resolving the Social Dilemma in Internet Routing Security Adoption Through Altruism, Bounded Rationality, and Network Externality 196
Keisuke Ishibashi and Tatsuya Oryu

Link Analysis Ranking

Predicting Interactions in the Weapons of Mass Destruction Knowledge Graphs 211
Abhigya Agrawal, Md Saidul Hoque Anik, and Ariful Azad

Backbone Extraction in Complex Networks via Embedding-Based Link Prediction 224
Jason Barbour, Ali Yassin, Hocine Cherifi, and Hamida Seba

Deep Learning Based Link Prediction for Minimizing Maximum Betweenness 237
Aashish, Parul Diwakar, Venkata Narendra Kotyada, Rishi Ranjan Singh, and Soumajit Pramanik

Global Link Prediction Strategies for Backbone Extraction in Real-World Networks 250
Ali Yassin, Hocine Cherifi, Hamida Seba, and Olivier Togni

Mobility and Networks

Kernel-Based Recovery of Traffic Graph Signals on Urban Networks 267
Rafael Alejandro Martínez Márquez and Giuseppe Patanè

Resilience and Disruption in Mexico's Intermunicipal Mobility Network: A Component Decomposition Approach 279
Erika Cruz-Bonilla, Issa Moussa Diop, Christian Wolff, Hocine Cherifi, and Maribel Hernández-Rosales

The Impact of Economic Factors on Brain Circulation Patterns 292
Afra Temiz, Mehmet Eren Bulut, and Mehmet Burak Akgun

Physical Activity and Mental Health: A Comparative Analysis Across Age Groups Using Complex Networks 304
Rama Krishna Thelagathoti, Ka-Chun Siu, and Hesham H. Ali

Enhanced Vehicle Routing Algorithm for Reinforcement Learning Using Artificial Potentials for Multi-objective Logistic Network 316
Taku Iguchi, Kei Uchimura, Shigeyuki Miyagi, and Osamu Sakai

Democracy in the Diaspora: The Role of Politics in Migration Networks 328
Shayan Daijavad and Theresa Migler

Network Embedding

Topological Network Alignment Using Self-supervised Siamese Model 343
Aljohara Almulhim and Mohammad Al Hasan

Critical Synthetic Benchmarking of Graph Representation Learning 356
Noé Durandard, Camille Roth, and Telmo Menezes

Dissecting Embedding Method: Learning Higher-Order Structures from Data 369
Liubov Tupikina and Hritika Kathuria

Local Intrinsic Dimensionality for Dynamic Graph Embeddings 374
Dušica Knežević, Miloš Savić, and Miloš Radovanović

Urban Systems

An Application of Edge Betweenness Centrality in Bi-objective Optimization of Street Networks 389
Supharoek Chattanachot, Frédéric Guinand, and Kittichai Lavangnananda

Multiple Route Search Method on Road Network Using Genetic Algorithm with Virus Infection 402
Haruto Aoyagi and Takayasu Fushimi

Optimising Urban Electric Vehicle Charging Infrastructure: A Spatial Modelling Approach for Fleet Operators with Case Studies in Mumbai and Delhi 415
Sedar Olmez and Sachio Kobayashi

Road Network and its Impact on Urban Socio-ecological Systems: A Network Perspective 425
N. V. Neelima, Shuanglei Wu, Shaurya Shriyam, and Yongping Wei

A Complex Networks Approach to Evaluate the 15-Minute City Paradigm and Urban Segregation at Different Scales 435
Mirko Lai, Salvatore Vilella, Federica Cena, and Giancarlo Ruffo

Complex Network Analysis as a Tool for Road Maintenance Management 447
Annamaria Ficara, Giuseppe Sollazzo, Gaetano Bosurgi, and Maria Fazio

Author Index 459

About the Editors

Hocine Cherifi is a professor of Computer Science at the University of Burgundy, Dijon, France, since 1999. He completed his Ph.D. degree at the National Polytechnic Institute, Grenoble, in 1984. Before moving to Dijon, he held faculty positions at Rouen University and Jean Monnet University, in France. He has held visiting positions at Yonsei, Korea, University of Western Australia, Australia, National Pingtung University, Taiwan; and Galatasaray University, Turkey. His recent research interests are in computer vision and complex networks. He has published more than 200 scientific papers in international refereed journals and conference proceedings. He held leading positions in more than 15 international conference organizations (General chair, Program chair), and he served on more than 100 program committees. He is the founder of the International Conference on Complex Networks & their Applications. Currently, he is also a member of the Editorial Board of Computational Social Networks, PLOS One, IEEE Access, Journal of Imaging, Complex Systems, Quality and Quantity, and Scientific Reports. He is the Founding Editor-in-Chief of the Applied Network Science journal.

Luis M. Rocha is the George J. Klir Professor of Systems Science at the Thomas J. Watson College of Engineering and Applied Science, Binghamton University, where he leads the Complex Adaptive Systems and Computational Intelligence (CASCI) lab. He is also Senior Fellow at the Instituto Gulbenkian da Ciencia, co-director of the Consortium for Social and Biomedical Complexity between Binghamton University and Indiana University, and a founding partner of the International Center for Excellence in Mental Health Sciences. He has been the director of the NSF-NRT Interdisciplinary Training Program in Complex Networks and Systems, and Professor of Informatics at the Luddy School of Informatics, Computing, and Engineering at Indiana University. His research is on complex networks and systems, computational and systems biology, and computational intelligence. He received his PhD in Systems Science in 1997 from the State University of New York at Binghamton. From 1998 to 2004, he was a permanent staff scientist at the Los Alamos National Laboratory, where he founded and led a Complex Systems Modeling Team during 1998–2002 and was part of the Santa Fe Institute research community. He has organized major conferences in the field such as Alife X, ECAL 2007, and Complex Networks 2019–2023. He has published many articles in scientific and technology journals and has been the recipient of several scholarships and awards.

Chantal Cherifi received her PhD in Computer science from Corsica University, France, in 2011. Since 2014, she is working as an Associate Professor at the DISP laboratory, at the University of Lyon 2, France. Her main research interests are focused on information systems agility and big data management with applications on enterprise information systems and smart cities, using complex networks, ontologies, and Product Lifecycle Management (PLM) systems tools. She is involved in several international

conference organizations such as Complex Networks, CompleNet, PLM, and DICTAP. She serves as a member of international conferences program committees of Complex Networks, CompletNet, Complexis, ISCRAM-med, CSCESM, SITIS, and ICIEIS; and journal referee for IJCIM, EPL, Scientific Reports, and Nature. She is a member of EU Erasmus-Mundus programs (SmartLink and cLink program). Her local responsibilities include Committee Lab member, since 2016, and Lab seminar co-organizer, since 2014.

Murat Donduran is a Professor of Economics at the Department of Economics, YILDIZ Technical University Istanbul Türkiye, where he is the Director of the Graduate School of Social Sciences. He is also the Board member of in Turkish Economic Foundation. His research is on microeconomics, computational economics, firm dynamics, and game theory. He received his Bachelor's and Master's Degrees in Economics from Marmara University, Istanbul, Türkiye, and a Ph.D. in Economics in 2000 from the YILDIZ Technical University. Since 2000 he has been a faculty member in the Faculty of Economic and Administrative Sciences at the YILDIZ Technical University. He has organized several conferences in the field such as the YILDIZ International Conference on Social Sciences, the Annual International Conference on Social Sciences (2016–2020), and the International Conference on Economics (IceTea) 2019–2023. He has published many articles in scientific journals such as Games and Economic Behavior, Review of Industrial Organization, Physica A, and North American Journal of Economics and Finance.

Dr. Onur Varol is an Assistant Professor at the Sabanci University Faculty of Engineering and Natural Sciences and Principal Investigator at the VIRAL Lab. His research focuses on developing techniques to analyze online behaviors to improve individual well-being and address societal problems using online data. He is awarded by The Turkish Science Academy to Young Scientist Awards (2022) and Research Incentive Award by METU Parlar Foundation (2022). He also received TUBITAK 2247-D National Leader Researcher Grant in 2022. Prior to joining Sabanci University, he was a postdoctoral researcher at Northeastern University at the Center for Complex Network Research. He completed his PhD in Informatics at Indiana University, Bloomington (USA). His thesis focuses on the analysis of manipulation and threats on social media, and he was awarded the 2018 University Distinguished Ph.D. Dissertation Award. He has developed a system called Botometer to detect social bots on Twitter and his team ranked top 3 worldwide at the 2015 DARPA Bot Detection Challenge. His research published in prestigious venues such as International Conference of Web and Social Media (ICWSM), Nature Communications, Nature Human Behavior, World Wide Web (WWW) conference, and Communications of the ACM.

Biological Networks

SimNetX: Interactive Support for Biomedical Data Analysis Using Patient Similarity Networks

Tomas Anlauf[1], Kristyna Kubikova[1], Eliska Ochodkova[1](✉), Eva Kriegova[2], and Milos Kudelka[1]

[1] VSB - Technical University of Ostrava, Ostrava, Czech Republic
{tomas.anlauf,eliska.ochodkova}@vsb.cz
[2] Palacky University Olomouc and University Hospital Olomouc, Olomouc, Czech Republic

Abstract. Patient similarity networks (PSNs) are a widely used tool in biomedical data analysis. The most powerful feature of PSNs (and all networks in general) is the ability to visualize relatively complex data in a way that people can understand without expertise in statistics and machine learning. This aspect of data analysis is particularly important in biomedical data analysis, where data analysts, clinicians, and people from laboratories work in collaborative teams. However, working with PSNs requires steps that are usually solved in different systems or using programming libraries, e.g., in R or Python. This paper presents a tool whose design results from several years of experience with PSNs in biomedical data analysis. The tool focuses on the essential tasks of transforming vector data into PSNs and exploring them in detail, as well as on a high level of interactivity, e.g. allowing the formulation of preliminary hypotheses. We demonstrate the practical use of the tool on a well-known biological dataset.

Keywords: patient similarity network · biomedical data · data visualization

1 Introduction

In the field of healthcare, a substantial amount of patient data is acquired every day. These data are often complex in nature and present significant challenges in their analysis. Therefore, sophisticated advanced statistical methods and supportive computational tools are required to provide accurate and clinically meaningful results of these often heterogeneous data sources, such as patient-reported outcomes and genomic data. Multivariate analysis is one such method that has become increasingly important for its ability to process and analyze data with multiple variables simultaneously. These data often hold information about clinical measurements, lifestyle factors, genetic data, and imaging results. This approach is particularly useful in identifying factors that might affect disease risk, its progression, and response to treatment.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 3–14, 2025.
https://doi.org/10.1007/978-3-031-82439-5_1

In this paper, we present a tool SimNetX (SIMilarity NETwork eXplorer) to support the analysis of such data based on patient similarity network (PSN) exploration [12,16]. Patient records (vector data) are represented by vertices in the PSN and existing edges indicate the similarity between them. Similar patients are those, whose similarity exceeds a specified threshold, which is determined either manually or by network construction algorithm. PSNs are a powerful tool because they can be easily visualized using different layouts and allow the identification of subgroups with common clinical features or genetic markers, for example. There is not yet a methodology on how to properly use them to their full extent and what kinds of information we can extract from them.

The complexity of multivariate analysis of patient data in biomedical research often requires the participation of many experts and the use of different systems. In our paper, we present a tool for clinicians that simplifies the analysis process without requiring advanced technical knowledge. In designing it, we leveraged our experience with the complex multi-step process that we have observed while working with researchers in biomedical laboratories. By minimizing reliance on external data specialists, the tool aims to increase overall workflow efficiency and facilitate more effective data interpretation in the clinical setting. It allows researchers to interactively explore their data in the form of similarity networks and potentially generate initial hypotheses. Although the tool is designed for clinicians, this paper discusses technical background for network construction with its analysis and rather serves as the introduction of the system to the researchers with machine learning knowledge. We only briefly mention the methods we use in our tool as detailed description is out of the scope of this paper. The uniqueness and utility of this application stems from the fact that we have not yet found a similar system that is capable of converting vector data into a network structure with the ability to instantly explore and analyze it. Due to this fact, it was impossible for us to directly compare our tool to another similar system as they simply do not bundle network construction and its analysis together. The tool we present has already proved its worth in several cases, for example [11,12,19].

This paper is organized as follows. We first discuss similar systems that include PSN in their analysis capabilities and we summarize the possibilities of our tool. In the next section we will look at our tool and its functions in more detail. These are then demonstrated in the section dedicated to the analysis of the Ecoli dataset.

2 Related Work

PSNs are increasingly used in clinical research and precision medicine, providing a robust framework for analyzing and predicting patient outcomes, phenotypes, and disease risks by representing patients as nodes with their similarities as weighted edges. These networks integrate heterogeneous data types, including omics, clinical, laboratory, and imaging data, to offer a comprehensive view of patient relationships and enhance the explainability of machine learning models

[7]. Significant applications of PSNs include the integration of partial multimodal datasets, where certain data types may be missing for some patients. Algorithms employing message-passing learning strategies have shown effectiveness in recovering and integrating missing data, as shown with breast cancer data from The Cancer Genome Atlas (TCGA) [8]. PSNs are not only useful for patient profiling but also have applications in drug recommendation. For instance, the DAPSNet model leverages PSNs to encode multi-type medical codes and integrate drug information from similar patient states, mimicking human clinical decision-making and improving drug combination predictions [20]. Furthermore, PSNs are used in patient similarity searches, which is essential for personalized treatment plans. Methods using heterogeneous information networks (HINs) address challenges related to high-dimensional and sparse vector representations by connecting patients, diseases, and drugs while incorporating temporal information [9].

Effective utilization of PSNs in clinical settings also depends on visualization tools that can help researchers and clinicians explore patient data. Tools like MetaRelSubNetVis enable group-wise comparison of PSNs, enabling interactive exploration of patient-specific attributes [1]. This is in line with Van den Elzen and van Wijk's approach to network exploration, emphasizing the importance of transitions from detailed views to broader overviews through selections and aggregations, which are critical for making sense of multivariate network data [5]. Such functionalities are particularly valuable in clinical settings where effective communication and data sharing are critical.

Despite their potential, PSNs face challenges related to the complexity of network topology and the interpretability of similarity measures. Current classifiers often do not fully exploit the benefits of PSNs, leading to complex and underutilized networks. However, advancements like StellarPath, which integrates omics data hierarchically and utilizes graph convolutional neural networks, show promise in enhancing PSN utility by providing meaningful features and improving classification performance [6]. Additionally, the integration of deep learning techniques, including BERT, CNNs, and LSTM-based autoencoders, into PSNs has addressed challenges related to data heterogeneity and dimensionality, improving classification accuracy and preserving temporal features [14]. The netDx framework further exemplifies the application of PSNs for patient classification, outperforming traditional machine learning methods in cancer survival predictions and identifying significant biological pathways [17].

In this article we introduce a versatile visualization tool designed to enhance the analysis and interpretation of patient similarity networks, offering several advanced functions that set it apart from existing tools. Unlike other visualization platforms, which primarily focus on static representations or offer limited interactivity, our tool enables users to convert vector data into customizable network structures, allowing for the selection of specific attributes to create tailored networks. Among its key functions are advanced community detection algorithms and the ability to generate simple community statistics, including boxplots and Matthews correlation coefficient (MCC) calculations, which can

provide valuable insights into shared characteristics or disease patterns. Additionally, our tool provides flexible coloring options by attributes, communities, or classes, offering clearer visual insights than many existing tools that have limited coloring functionalities.

Beyond visualization, our tool supports a wide range of data manipulation techniques. Users can apply various filtration methods to refine the network and perform attribute transformations such as normalization, scaling, standardization, and conversion from log-normal to normal distributions, functionalities often restricted or unavailable in other tools. To further support data exploration, the tool includes histogram generation for visualizing attribute distributions, making it a well-rounded solution for the detailed analysis and visualization of PSNs.

3 Tool Overview

To achieve the goals mentioned earlier, the tool integrates several functionalities that support clinicians in managing and analyzing datasets. These functionalities are designed to provide user-friendly methods for data exploration, transformation, and visualization, enabling effective data analysis without requiring advanced technical expertise. The diagram in Fig. 1 describes the key functions of the tool divided into sections, which will be described next.

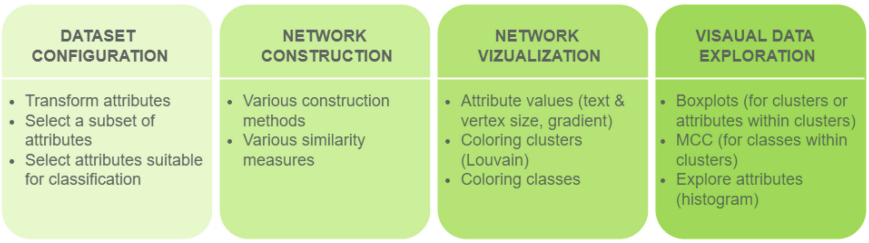


Fig. 1. Overview of key tool functions.

3.1 Dataset Configuration

The tool provides several functions to assist users in preparing and transforming datasets for analysis. Users can explore the distribution of individual attributes by generating histograms, which visualize data patterns and help detect outliers. The tool also offers various options for attribute transformation, including normalization, rescaling, logarithmic transformation, and standardization. To simplify the dataset, users can select a subset of attributes, allowing them to focus on the most relevant data for their analysis. The tool also enables users to manually choose attributes for classification tasks based on their own knowledge

Fig. 2. The tool's GUI overview: 1. Layout, coloring, remodeling settings panel; 2. Attribute transformation options; 3. List of active attributes used for network construction; 4. Network canvas; 5. Numerical attribute histogram and value for selected node (red bin shows the location of value of node 299 in the distribution); 6. Selected node's neighbour panel.

and understanding of the data. All the functions for dataset configuration can be found in Fig. 2, where the panel 1 is displaying the overall configuration options, panel 2 shows the attribute transformation options and panel 3 the attribute selection.

3.2 Network Construction

One of the most important functions provided by our application is the ability to transform vector data into a network structure. This eliminates the need for a third party tool or manual processing by an expert in the field. We have not found a similar system that can construct a network followed by the ability to start analyzing it immediately. This gives the user power to adjust the construction process and check if the outcome is suitable for further exploration. At the beginning of this process, the tool calculates chosen $N \times N$ similarity matrix S, where S_{ij} represents the similarity value between numerical attribute value vectors of nodes i and j, and N is the number of nodes. The tool supports several similarity measures: for unrestricted numerical values it is the Gaussian kernel, cosine similarity, and the absolute value of Pearson or Spearman correlation. For binary values it uses Jaccard index and co-occurence. A construction algorithm then uses this matrix to decide whether to form a link between nodes. The tool allows users to choose from two construction algorithms. The first is Epsilon kNN [18], where the similarity threshold and minimal number of neighbours

is set by the user and a pair of nodes forms a connection if their similarity is above the threshold. The second option is the LRNet algorithm [15], that is a semi-automatic network construction algorithm that calculates a local property called representativeness, which ultimately determines how many neighbours will a node $n_i \in N$ have. Nodes with the highest similarity to n_i are favoured for becoming its neighbor.

3.3 Network Visualization

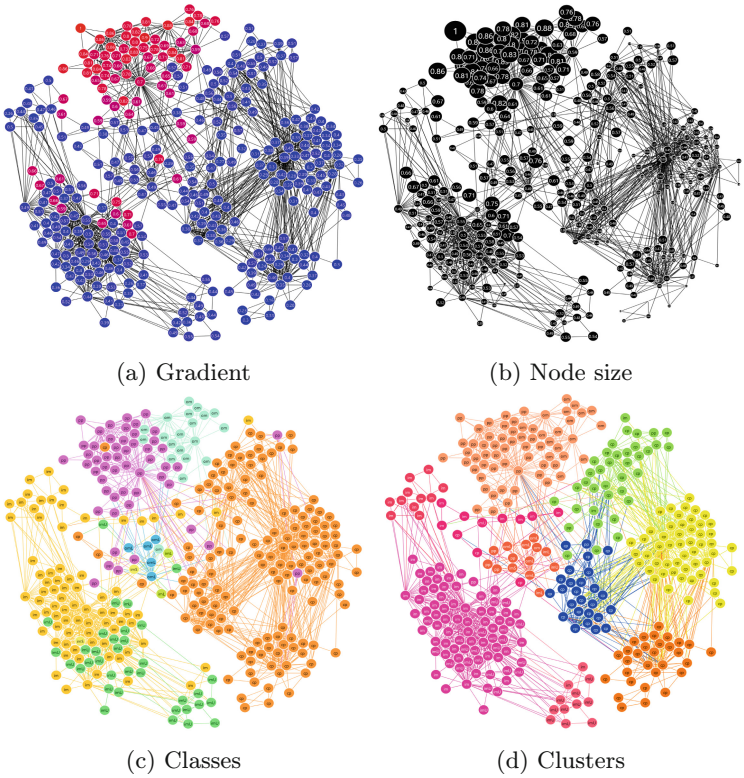
Another key function of our tool is its ability to visualize networks using various color-coding methods (Fig. 2 panel 4). The layout is computed using the ForceAtlas2 [10] algorithm, which arranges nodes based on their relationships, creating a spatial representation that reflects the network's structure. Attribute values can be displayed using color gradient or node sizes and text label sizes (Fig. 3a 3b), helping users quickly grasp key information. Beyond individual attributes, the tool also supports the coloring of clusters within the network, which are identified using the Louvain method (Fig. 3d). Users can also apply color coding to specific classes, allowing for a clear differentiation of categories or groups based on their classification (Fig. 3c). With these customizable visualization techniques, the tool allows users to explore and interact with their data in a more intuitive and insightful way.

3.4 Visual Data Exploration

As previously mentioned the tool also provides automatic community detection within the network using the Louvain algorithm [2]. This algorithm identifies clusters by optimizing the modularity of the network, allowing for clear separation of densely connected groups. By detecting these communities, the tool enables further analysis alongside predefined classes, providing insights into their interactions.

Users can also explore the distribution of attribute values using histograms (Fig. 2 panel 5) and visualize relationships within clusters using box plots, either for the clusters themselves or for specific attributes within clusters. Additionally, users can display the neighbors of a selected node to better understand its local network context (Fig. 2 panel 6). To evaluate the accuracy of the classifications within the detected communities the tool incorporates Matthews Correlation Coefficient (MCC) [4], which offers a balanced evaluation of the relationship between clusters and associated classes. MCC was chosen for its ability to work well with unbalanced data, which is typical in biomedicinal field. There are several levels of how well the cluster correlates with ground-truth class [3]: 0.2 moderate, 0.4 good, 0.7 strong, 0.9 excellent. MCC is calculated for cluster c and class cl as follows:

$$MCC(c, cl) = \frac{(TP * TN) - (FN * FP)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (1)$$



(a) Gradient (b) Node size

(c) Classes (d) Clusters

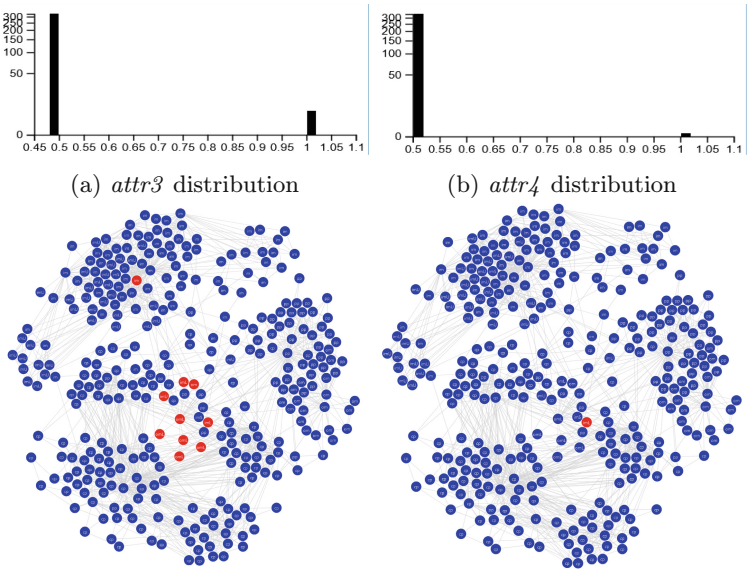
Fig. 3. Network coloring.

where TP is the number of nodes in the cluster c with class cl, FP is the number of nodes with in c with any class other than cl, TN is the number of nodes outside of c with any class other than cl and FN is the number of nodes outside of c with class cl.

4 Proof of Concept

In the following section, we demonstrate the use of our tool to analyze the well-known Ecoli dataset [13], which contains information about protein localization sites. It consists of 336 instances and 7 attributes, divided into 8 unbalanced classes. The dataset contains only numerical attributes, along with id and class columns, and has no missing data. No preprocessing was required. This analysis serves as a proof of concept for our tool; extending the demonstration to other datasets falls outside the scope of this paper.

The first step of the analysis is the construction of the network. After uploading the dataset file, we select a conversion algorithm with a similarity measure. Once the tool generates the network, the user is free to remodel its structure



(a) *attr3* distribution (b) *attr4* distribution

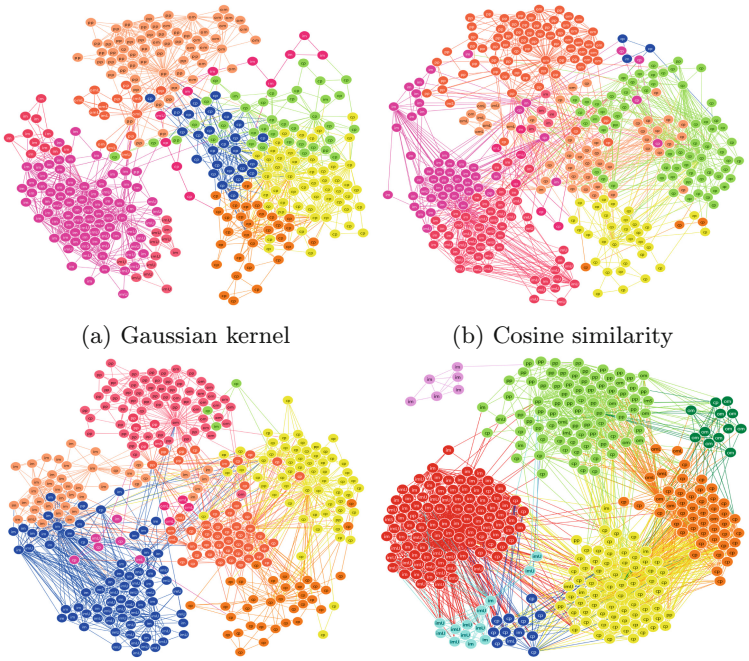
(c) *attr3* gradient (d) *attr4* gradient

Fig. 4. Distribution of attribute values displayed in histograms and by gradient coloring in the network.

by experimenting with different conversion algorithms, similarity measures, and attributes to be included in the calculation. By inspecting the initial constructed network together with attribute histograms, we noticed two attributes (*attr3*, *attr4*) where most nodes have the same value with a few outliers (Fig. 4a, 4b). If we change the network's color to gradient mapped to attribute values (Fig. 4c, 4d), we can see that all instances of class *omL* have high values for *attr3*. Only one node has a high value for *attr4* which we consider an outlier. Therefore, we decided to keep *attr3* for network construction, as it may be a valuable attribute for distinguishing class *omL* from the other instances, and exclude *attr4* to reduce potential noise. The resulting networks constructed with different similarity measures (ABS for absolute value) are shown in Fig. 5.

We decided to continue our exploration with the Spearman correlation network, as it seems to separate the classes more effectively than the other similarity measures. How well the clusters correspond with the classes can be determined by examining the MCC values calculated for each class in each cluster in Fig. 6. We may observe that the vast majority of clusters have one dominant class with $MCC > 0.4$, which is considered a good result.

We were also curious about why the network was constructed in this particular way, leading to specific cluster detection. While inspecting the attribute boxplots, we identified two particular attributes (Fig. 7) with distinct medians in many clusters, suggesting that these attributes were essential to the resulting network structure. Additionally, two clusters (orange and yellow) primarily



(a) Gaussian kernel (b) Cosine similarity

(c) ABS of Pearson correlation (d) ABS of Spearman correlation

Fig. 5. Resulting networks constructed with different similarity measures. Colors represent detected clusters.

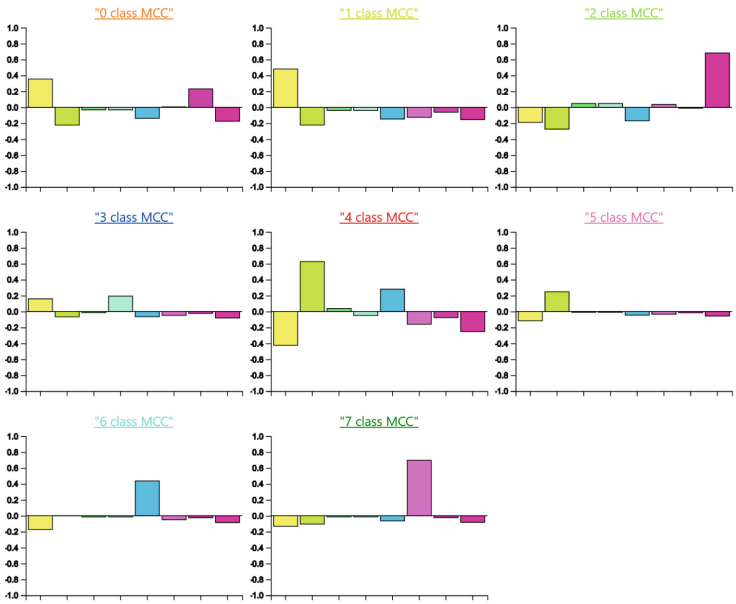


Fig. 6. Matthews correlation coefficient for all classes in each cluster. Each bin color represents a different class.

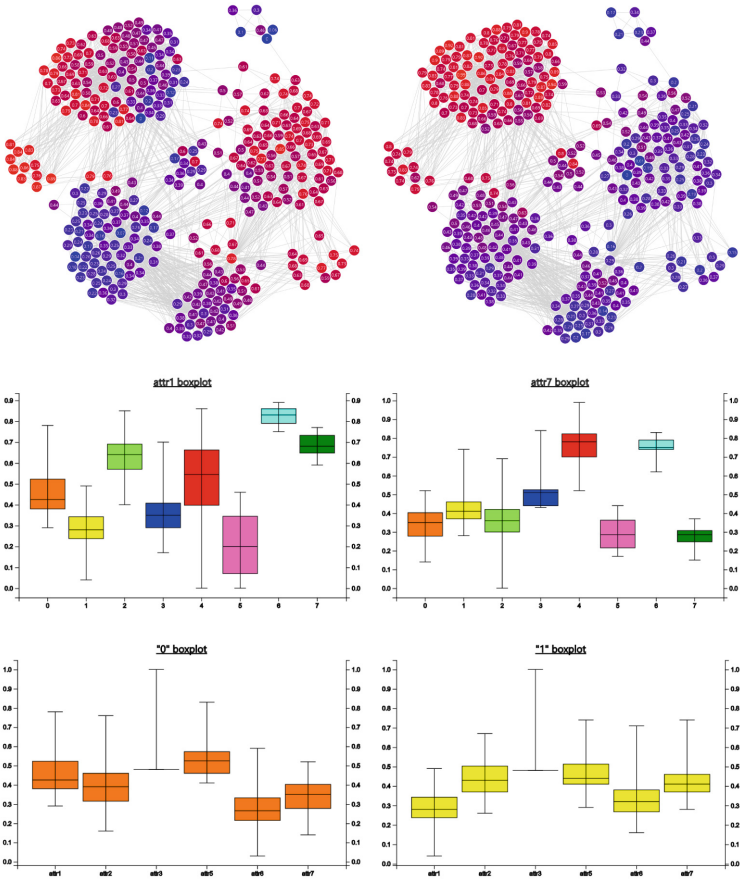


Fig. 7. Important attributes exploration and statistics. Gradients network coloring of attribute *attr1*, *attr7* (top). Boxplots of *attr1*, *attr7* in every cluster (middle). Boxplots for all attributes in clusters 0 and 1 (bottom).

consisted of the same class, raising the question of why they were separated. Boxplots for each attribute in these clusters revealed slight differences between attribute values, with *attr1* being the most prevalent. This distinction explains the detection of two separate clusters and suggests that *attr1* is indeed the most important attribute. Excluding it would significantly degrade the quality of the clustering.

5 Conclusion

In this paper, we have discussed patient similarity networks in biomedical data analysis. To the best of our knowledge, there is no interactive software tool that supports the construction of a network from vector data and provides the visualized outputs necessary to interpret the results. The presented tool, SimNetX, is

designed to achieve these goals and is the result of several years of collaboration in a multidisciplinary team involved in biomedical data analysis. We believe the primary advantage of this tool lies in its interactivity and no-code approach, that allows researchers without advanced technical knowledge to analyze their data effectively. So far we have received a positive feedback from the clinicians and researchers we are working with and in the future, we anticipate further development based on feedback from users who choose to take advantage of the tool's capabilities. In the current version, most of the data and configuration settings are in the hands of the user and the outputs may not always be useful. Therefore, the biggest challenge is to support automatic or semi-automatic network generation, which will speed up the discovery of networks with analytically and domain usable properties.

Funding Information. This work was supported by SGS, VSB-Technical University Ostrava, under grant no. SP2024/007 and by MH CZ - DRO (FNOl, 00098892).

References

1. Auer, F., Mayer, S., Kramer, F.: Metarelsubnetvis: referenceable network visualizations based on integrated patient data with group-wise comparison. bioRxiv, pp. 2022–04 (2022)
2. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Statist. Mech. Theory Exp. **2008**(10), P10,008 (2008)
3. Chicco, D., Starovoitov, V.V., Jurman, G.: The benefits of the Matthews correlation coefficient (MCC) over the diagnostic odds ratio (DOR) in binary classification assessment. IEEE Access **9**, 47112–47124 (2021)
4. Chicco, D., Tötsch, N., Jurman, G.: The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. Biodata Mining **14**(1), 13–13 (2021)
5. Van den Elzen, S., Van Wijk, J.J.: Multivariate network exploration and presentation: from detail to overview via selections and aggregations. IEEE Trans. Visual Comput. Graph. **20**(12), 2310–2319 (2014)
6. Giudice, L., Mohamed, A., Malm, T.: Stellarpath: hierarchical-vertical multi-omics classifier synergizes stable markers and interpretable similarity networks for patient profiling. PLOS Comput. Biol. **20**(4), e1012,022 (2024)
7. Gliozzo, J., et al.: Heterogeneous data integration methods for patient similarity networks. Brief. Bioinform. **23**(4), bbac207 (2022)
8. Gliozzo, J., Patak, A., Gallardo, A.P., Casiraghi, E., Valentini, G.: Patient similarity networks integration for partial multimodal datasets. In: BIOINFORMATICS, pp. 228–234 (2023)
9. Huang, H., Lu, X., Guo, W., Jiang, X., Yan, Z., Wang, S.: Heterogeneous information network-based patient similarity search. Front. Cell Dev. Biol. **9**, 735,687 (2021)
10. Jacomy, M., Venturini, T., Heymann, S., Bastian, M.: Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. PloS one **9**(6), e98,679 (2014)

11. Janca, O., Ochodkova, E., Kriegova, E., Horak, P., Skacelova, M., Kudelka, M.: Real-world data in rheumatoid arthritis: patient similarity networks as a tool for clinical evaluation of disease activity. Appl. Netw. Sci. **8**(1), 57 (2023)
12. Mikulkova, Z., et al.: Deciphering the complex circulating immune cell microenvironment in chronic lymphocytic Leukaemia using patient similarity networks. Sci. Rep. **11**(1), 322 (2021)
13. Nakai, K.: Ecoli. UCI machine learning repository (1996)
14. Navaz, A.N., El-Kassabi, H.T., Serhani, M.A., Oulhaj, A., Khalil, K.: A novel patient similarity network (PSN) framework based on multi-model deep learning for precision medicine. J. Personal. Med. **12**(5), 768 (2022)
15. Ochodkova, E., Zehnalova, S., Kudelka, M.: Graph construction based on local representativeness. In: Cao, Y., Chen, J. (eds.) COCOON 2017. LNCS, vol. 10392, pp. 654–665. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62389-4_54
16. Pai, S., Bader, G.D.: Patient similarity networks for precision medicine. J. Mol. Biol. **430**(18), 2924–2938 (2018)
17. Pai, S., Hui, S., Isserlin, R., Shah, M.A., Kaka, H., Bader, G.D.: netdx: interpretable patient classification using integrated patient similarity networks. Mol. Syst. Biol. **15**(3), e8497 (2019)
18. Silva, T.C., Zhao, L.: Machine Learning in Complex Networks. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-17290-3
19. Sova, M., et al.: Network analysis for uncovering the relationship between host response and clinical factors to virus pathogen: lessons from SARS-COV-2. Viruses **14**(11), 2422 (2022)
20. Wu, J., Dong, Y., Gao, Z., Gong, T., Li, C.: Dual attention and patient similarity network for drug recommendation. Bioinformatics **39**(1), btad003 (2023)

Class Dominancy Profiles in Multi-class and Multi-cluster Similarity Networks

Ondrej Janca[1(✉)], Arootin Gharibian[1], Eliska Ochodkova[1], Eva Kriegova[2,3], and Milos Kudelka[1]

[1] VSB - Technical University Ostrava, Department of Computer Science, Ostrava, Czechia
ondrej.janca@vsb.cz
[2] Palacky University Olomouc, Olomouc, Czechia
[3] University Hospital Olomouc, Olomouc, Czechia

Abstract. The visual representation of data analysis results, a current trend in data science, is essential for people without deep knowledge of statistics and machine learning. Patient Similarity Networks (PSNs), as an instance of similarity networks in general, are commonly used visualization tools in biomedical data analysis. PSNs are an understandable presentation of the complex relationships hidden in patient data, such as partitioning patients with similar characteristics into clusters, providing relatively easy interpretation to clinicians and clinical biologists. However, the interpretation may be inaccurate due to an incorrect intuition, especially in complex multi-class, multi-cluster situations. Our paper focuses on analyzing cluster-class relationships in PSNs, assuming different numbers of classes and clusters, both of different and potentially imbalanced sizes. We use several problematic situations to show how to support clinical intuition in interpreting these situations correctly. We use the Matthews correlation coefficient to analyze cluster-class relationships in a way that understandably complements the PSN visualization. Finally, we present a visualization of this approach on a real-world patient similarity network.

1 Introduction

Real-world data often exhibit challenging characteristics, such as non-normal distribution of attributes, dubious reliability, missing values, high dimensionality, and heterogeneity. The attributes are frequently soft data and are often transformed into ordinal scales to simplify data collection, which further complicates data analysis. Since the patterns and trends in real-world clinical data tend to be weak, distorted, and noisy, classical statistical approaches are often insufficient to draw correct conclusions.

There are numerous methods for data analysis, and it can prove challenging to decide which to employ to process the data correctly. However, a common goal of these methods is to find the information hidden in the data in a way that

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 15–26, 2025.
https://doi.org/10.1007/978-3-031-82439-5_2

selectively reduces data complexity, preserves patterns and relationships therein, and is understandable and explainable in a practical manner.

A robust method for real-world data analysis is a combination of similarity networks and community detection (clustering). Vector data of objects in the dataset are used to measure the objects' similarity, which is then used to construct a network. In this network, the vertices represent the objects in the dataset and weighted edges represent the similarity between them. This approach balances complexity reduction and pattern retention [13]. A community detection (clustering) follows and detected communities (clusters) are then investigated in relation to classes (or, in general, any categorical labels). Detected relationships and trends are then well explainable by a visualization.

Patient similarity networks (PSN), as an instance of similarity networks in general, are a commonly used tool for biomedical data analysis. In a PSN, the objects of the dataset represent individual patients, and the edges between them show their similarity. Previously introduced LRNet algorithm [14] constructs a PSN based on local representativeness and Gaussian kernel similarity, and patient clusters are formed utilizing the Louvain algorithm for community detection [2]. Later works showed its effectiveness in analyzing biomedical data [16,17]. The analyzed network is then visualized (Fig. 1), offering easy explainability and understandability to clinicians. This makes it a valuable tool for communication between data scientists and clinicians.

Although visualizing the data as a clustered network is a convenient and accurate representation of a dataset, changes in the PSN construction attributes will lead to different networks. This is an expected outcome since there is no "best network" to describe the relationship of all attributes of any dataset. Thus, after a network visualization is prepared, it is necessary to evaluate its quality. In a previous work [10], we presented a methodology for this evaluation. Another key aspect of the PSN analysis is the evaluation of the relationships between detected patient clusters and classes (categorical labels).

This paper presents a method for visualizing the evaluation and measurement of cluster-class relationships in PSNs. First, we briefly discuss the current approaches to evaluate cluster-class relationships and their explainability for non-statisticians. Then, we discuss the cluster-class relationship and its interpretability for real-world dataset analysis. Then, we explain our approach of using the Matthews Correlation Coefficient (MCC) to assess the dominancy rank of individual classes in each cluster of any PSN. Finally, we demonstrate in several synthetic examples and one real-world dataset how this approach can be visualized and help clinicians make informed decisions.

2 Cluster-Class Relationship

The relationship between a cluster and a class can be viewed from multiple angles, answering different questions. The evaluation of cluster-class (cluster-label) relationships is often based primarily on the expertise of the data analyst and the preference and domain knowledge of the clinician. However, the discussion between a data analyst and a clinician can be difficult. For example, the

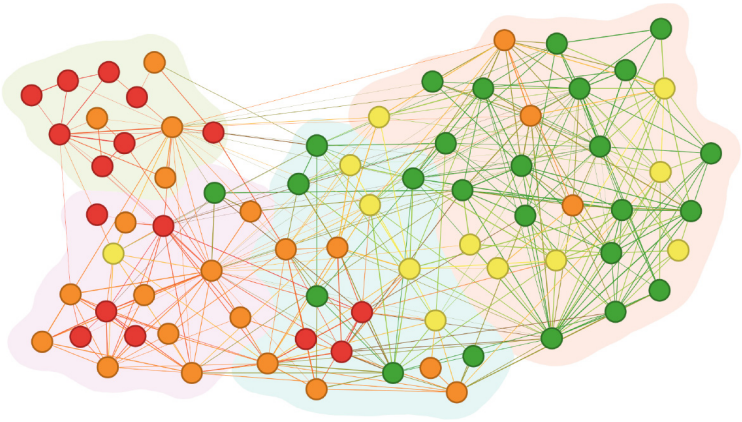


Fig. 1. A real-world PSN. This is an example of a real-world Patient Similarity Network comprising four clusters of rheumatoid arthritis patients [10]. The four classes represent the increasing severity of the disease (in order: green, yellow, orange, and red).

clinician can be interested in one good cluster that is well related to a class, even though the rest of the dataset remains in bad clusters or is a random noise. Additionally, the real-world data scarcely results in ideal networks and clustering solutions, but the expectation for an ideal case (all classes in individual neat clusters) is substantial. Our method of measuring the relationship between clusters and classes aims to explain complex and biased situations encountered in real-world data.

Based on our experience from collaboration with clinicians, the information in clustering solutions of medical datasets can be condensed into the following questions:

1. How good is the cluster in terms of individual classes?
2. How informative is a cluster in the context of the other clusters?

Different methods can provide answers to these questions; however, getting answers in a multi-class, multi-cluster (with two or more classes and two or more clusters) imbalanced dataset is challenging. The first question inquires about the number of objects both within and outside the cluster. The second question broadens the interest in how the objects outside of the cluster are distributed. Are they grouped in other good clusters? Are they randomly distributed? The method to measure this must be both informative and well-explainable to the clinician. Where a quality measure would be an ultimate tool for a data scientist, a visualization helps the clinician to understand the cluster and class relationship more easily. This makes the visualization component of the analysis vital for collaboration with clinicians. More importantly, it helps the clinician assess the cluster in the context of the dataset.

In binary cases (two classes and two clusters), a numeric expression of the evaluation is trivial with well-established measures: Accuracy, F-measure, and

methods based on the Receiver operator characteristics curve (ROC) are the most widely used measures. Not all of these measures consider the complete confusion matrix, though. Previous works show that the performance of Matthews Correlation Coefficient [12], which notably does consider the True Negatives, is more robust [6,7]. Despite reports on its lower performance when used in imbalanced datasets [5,11], MCC still outperforms other confusion matrix-derived measures [3,4].

However, there is another downfall to MCC: it is designed for a binary evaluation. There are extensions of MCC for multi-class, multi-cluster cases, namely R_K-statistic [8] and Extended Matthews Correlation Coefficient (eMCC) [1], but these require a symmetrical number of classes and clusters. This problem is solved for measures like Accuracy and F-measure, where micro- and macro-averaging are profoundly explored [15]. Even though this can be applied using MCC in the form of R_K-statistic and eMCC in limited cases mentioned above, this approach combines the information about individual clusters into one number, effectively obfuscating important details about the dataset.

These facts, however, do not prohibit the use of MCC in a one-vs-all evaluation, where the false positives and true negatives are the objects of any other class than the evaluated one. This way, the problem of multi-class evaluation can be converted to a set of binary problems for individual classes, and the clusters can be evaluated based on the MCC of each class.

3 Class Dominancy

As discussed in the previous section, when building a patient similarity network and subsequently detecting communities of patients, it is crucial to measure the quality of a cluster concerning both classes and other clusters. The most intuitive assessment considers the prevalence of a class in a cluster. However, we argue that in the context of real-world data, there is another critical aspect: the importance of a cluster regarding classes that are less prevalent or outright non-present in the cluster in question. We call this relationship Class Dominancy, and we assess it by measuring the MCC of individual classes.

MCC for class C in cluster L is formally defined as

$$MCC = \frac{TN \times TP - FN \times FP}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{1}$$

where TN are True Negatives (objects of other classes outside of cluster L), TP are True Positives (objects of class C within cluster L), FP are False Positives (objects of other classes within cluster L), and FN are False Negatives (objects of class C outside of cluster L).

The MCC values span the interval $[-1, 1]$ where 0 represents a completely random case. Generally, $MCC > 0.4$ is understood as a strong positive relationship, and $MCC > 0.7$ as a very strong positive relationship. In clinical medical data, these two thresholds signify a "good match" and a "very good match" between a cluster and a class, respectively. Inversely, $MCC < -0.4$

and $MCC < -0.7$ signify a "bad match" and a "very bad match" respectively. Note that this interpretation is domain-specific, and in predominantly soft-data domains (e.g., psychology), the thresholds are typically less strict.

Let us define the *Dominant class* D as one of the classes $C_1, ..., C_n$ in cluster L with the highest MCC value for cluster L. Note that on rare occasions, there may be multiple dominant classes in a cluster, especially in the case of small networks. In such a situation, it is advisable to inquire more deeply about the cluster and classes in question.

The Dominant class answers the question of which class present in the cluster is the most important one for the evaluation of said cluster. Since MCC cannot be directly averaged like F-measure and other measures, let us informally define a "good" cluster as such that fulfills two conditions: one class in this cluster has a high value of MCC (the dominant class), and other classes have low (preferably negative) values of MCC.

The first condition ensures that the cluster represents one class well. The other condition represents the inverse relationship for other classes, i.e., the cluster represents other classes badly. This further emphasizes the "goodness" of the relationship with the dominant class. In real-world data, we scarcely encounter clean and noise-free datasets. It is thus essential to emphasize the importance of a cluster that distinguishes itself from this ever-present noise. This is further discussed in Subsect. 4.4.

To provide clinicians with an easy understanding, MCC values of individual classes in the cluster of interest are then visualized as a column chart. We call this visualization a Class Dominancy Profile.

Since this method is based on MCC, it considers the complete confusion matrices for individual classes, thus providing critical information about the complete picture. This information would be otherwise lost if (averaged) $F_1 - score$ or similar measures were used. On the other hand, it was reported [9] that MCC cannot be averaged reliably for imbalanced data, that is, outside of problems with an equal number of clusters and classes suited for use of R_K-statistic [8]. Thus, the individual classes must first be measured individually by MCC and then evaluated by other means. For this, we use a visualization of MCC for individual classes. In the following section, we present examples of clustering situations, which are generally difficult to evaluate, and demonstrate how our methodology explains them.

4 Clustering Problems

The following synthetic problems are based on our experience with PSNs of real-world medical data. The examples presented here are simplified from networks to clusters of objects without edges. However, each object should be perceived as representing tens or hundreds of patients. The classes then represent diagnoses or distinct sub-groups of disease patients (e.g., requiring different treatment). The methodology presented here is intended to be objective, and thus, all diagnoses are considered equally important. If individual diagnoses (classes) have different

importance, for which there may be various clinical reasons, this non-objectivity must be carefully taken into account by the clinician when interpreting the MCC values and clusters. In each example below, we compare a clinician's intuition about evaluation to the information provided by our method.

4.1 Example 1 - Good Clusters

The model situation presented in Fig. 2 presents a well-clustered solution. Note that the dataset is imbalanced (clusters and classes have different sizes), and the clusters contain noise. This noise, however, never achieves $MCC > 0$. Each cluster contains one dominant class that displays a very good match with the cluster ($MCC \geq 0.7$). Other classes present in the clusters display a negative match.

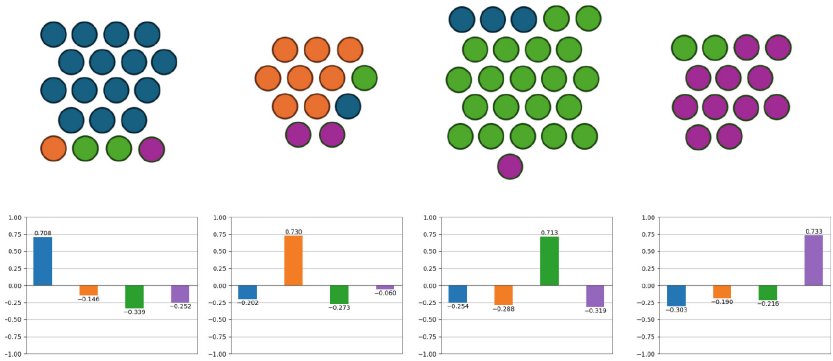


Fig. 2. Example of good clusters. The four clusters are considered "very good" since each contains a single dominant class with $MCC \geq 0.7$ and other classes present have $MCC < 0$.

This is a typical example of an ideal situation expected in a setting such as a clinical trial, as would be easy to interpret by a biomedical scientist or a clinician. Most of the subjects of a specific class belong in a single cluster, together with only a minority of objects of other classes. In this example, the prevalence and dominancy agree, and a clinician can understand the situation clearly.

4.2 Example 2 - Dominant Class with Low Prevalence

A model situation where a dominant class of a cluster is not the most prevalent class in the cluster is shown in Fig. 3. A clinician may intuitively consider the rightmost cluster to be a good cluster since it only captures one class (green), and the middle cluster is enriched with the orange cluster. However, in this example, although the blue class is not the prevalent class of the leftmost cluster, it is the dominant class, and the clinician should consider this cluster as a cluster to assess the blue class.

Interestingly, the dominancy of the blue class in the leftmost cluster is higher than that of the green class in the rightmost cluster, which may be counterintuitive. A clinician would also observe that the first cluster represents a very good sensitivity test for the blue class but a terrible specificity test for the same. This is in accord with the intuition described above since sensitivity is usually the first-line preference in medical testing (the idea is to detect maximum possible positives with a cheap, highly sensitive test and filter out false positives with a highly specific and often more expensive test later).

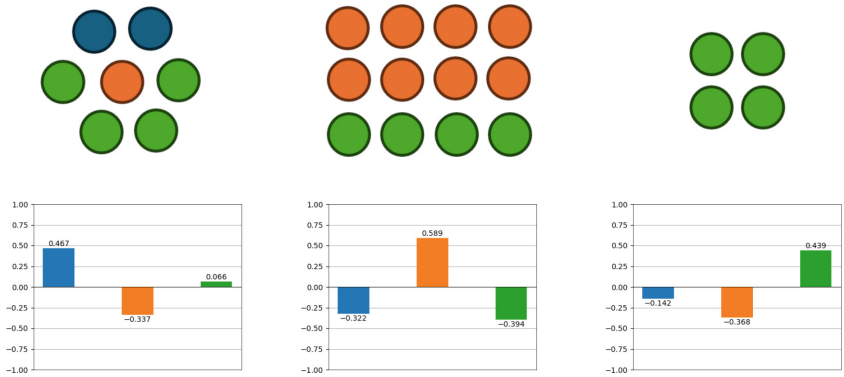


Fig. 3. A dominant class with low prevalence.

The emphasis a clinician would put on class prevalence is misleading. Let us consider the first cluster. Despite the low prevalence of the blue class, it still has the highest MCC in the first cluster. This is caused by the fact that this cluster captures all objects of the blue class. In terms of measures, the Recall or Sensitivity for the blue class are 1. The dominant class is the one that warrants the greatest focus during evaluation.

The clinician's intuition about the second cluster is correct, as is confirmed by the MCC value of the orange class. One could argue that with the prevalence of only 67%, the MCC value is counterintuitively high. Note, though, that the green class is spread equally among the clusters. Thus its influence in the second cluster is limited by the cluster's low recall for the green class.

The third cluster is neat for the green class, but a limiting factor is the aforementioned low recall. Despite that, the MCC still suggests a good relationship with the green class because of the high specificity. If a clinician evaluated this situation, they would put more emphasis on the Prevalence of classes in the clusters. This intuition is nevertheless misleading. The dominant class is the one that is most influential within the cluster and should be most carefully scrutinized during evaluation.

4.3 Example 3 - A Class Dominant in Multiple Clusters

In the model situation in Fig. 4, the blue class is dominant in two clusters. This may arise in situations where the objects of a class form two distinct groups. This previously unknown heterogeneity within the class should always invoke further scrutiny by a clinician. It is possible that the class (label) lacks resolution and that these two subgroups may behave differently in the clinical setting (e.g., react differently to treatment).

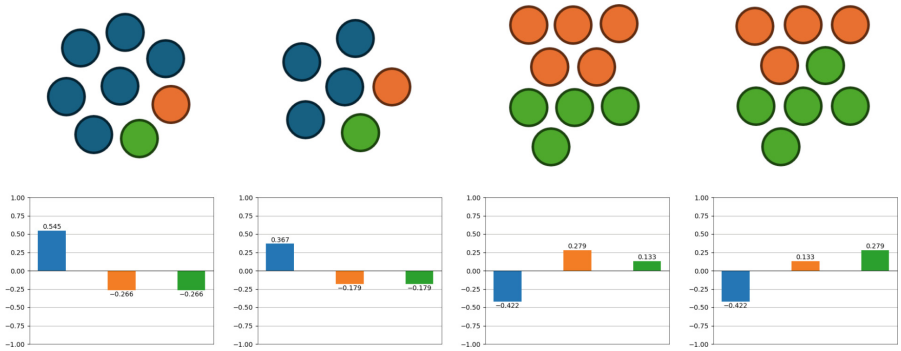


Fig. 4. A class dominant in multiple clusters.

In this example, a clinician would notice two distinct clusters enriched with a good dominancy with one class (blue) and two clusters with the other two classes. However, the distribution of the green and orange classes is near random if the dominancy of the classes is considered. Consequently, the clinician may dismiss the network altogether and look for alternative PSNs for better cluster-class (cluster-label) matching.

4.4 Example 4 - Influence of Random Noise and Multiple Dominancy

In this case, a clinician might find both clusters in Fig. 5A and 5B mildly informative regarding the blue class. The clinician's intuition may underestimate the random case since it is not profoundly emphasized within the clinical setting. The distribution of other classes in Fig. 5A is close to random, as indicated by their close-to-zero MCC values. In Fig. 5B, although the prevalence of the blue class in the left cluster is the same, the MCC value of the orange class is much closer to that of the blue class. Thus, the information concerning the dominant class is not as pronounced as in Fig. 5A. The second of two characteristics of a good cluster mentioned in Sect. 3 (about how the non-dominant classes should have low MCC values) is not met here. This shows how the easy explainability of the method can rapidly enhance the clinician evaluation.

Note the other two classes, green and purple. In example 5A, the two classes, together with the orange class, are distributed almost randomly. Here the intuition might lead a clinician to the conclusion that they are better distributed into the left cluster, but note that the cluster is bigger. If one accounts for this bigger size, it is apparent that the distribution is still close to random.

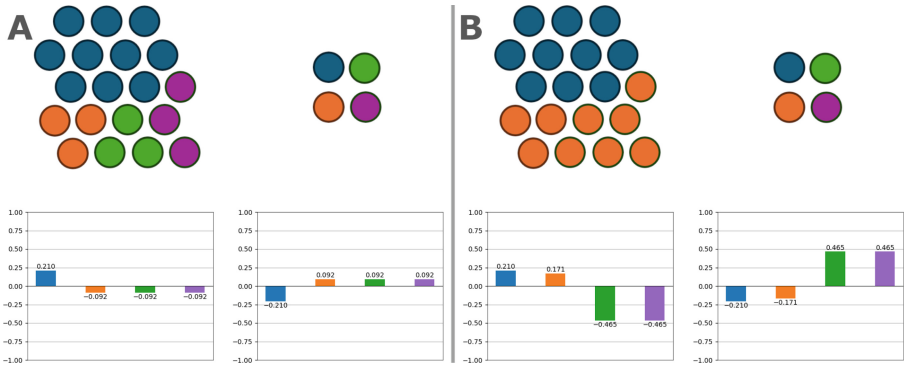


Fig. 5. Influence of random noise on the evaluation.

This, however, is different in example 5B. Here, the green and purple classes are only present in the smaller cluster on the right. The intuition might lead to the conclusion that the distribution of classes in the right cluster is random, but this is not true. The blue and the orange classes are mainly distributed in the larger cluster on the left. Taking this into account, it is apparent that their presence in the small cluster is mainly a noise. Also, the two other classes, green and purple, are present solely in this cluster. In this situation, a clinician might want to investigate the similarity of these two classes. Should they be dissimilar, they would probably form two separate small clusters. In this situation, though, they cluster together. This fact suggests that their similarity is high, and the clinician should consider merging these two classes together, should that make sense from the clinical point of view.

5 Application on Real-World Data

The methodology demonstrated in this paper can be implemented directly into PSN visualizations with Class Dominancy Profiles. This allows for a fast and easy evaluation of the network and expedites the process of PSN analysis. In our previous work [10], we focused on creating the network and analyzing the attribute trends within the network. We presented the PSN of rheumatoid arthritis (RA) patients (Fig. 1). In the previous work, we also characterized the clusters by the prevalence of active or inactive patients (classes of disease activity), measuring their precision (P), recall (R), and F_1-score (F_1). The four classes of disease

activity were merged into only two (active and inactive). This decision was supported by both the clinical guidelines for the treatment of RA patients and by the data, as the four-stage resolution proved too high to present understandable output.

Here, we use Class Dominancy Profiles (Fig. 6) to characterize these clusters more precisely, using the original four activity classes. To allow an easy assessment for the clinicians, we emphasize two important thresholds of $MCC > 0.4$ and $MCC > 0.7$ by horizontal lines (red and green, respectively).

From the new visualization, it is instantly apparent that the olive cluster on the left of the network is well related to the red class. The orange class in this cluster corresponds to a random case, and both green and yellow classes have negative MCC values. We can thus consider this cluster "good" which corresponds with the original evaluation ($P = 1$, $R = 0.31$, $F_1 = 0.47$).

The purple cluster on the left of the network shows a trend for dominancy of the orange class, but this does not reach the required threshold. This evaluation is harsher than the one in the original work ($P = 0.88$, $R = 0.38$, $F_1 = 0.54$).

The light blue cluster in the lower part of the network is, according to the Class Dominancy Profile, almost entirely random. This is more apparent from the visualization, compared to the original evaluation ($P = 0.53$, $R = 0.27$, $F_1 = 0.36$).

The orange cluster on the right, however, shows a good relationship with the green class, slightly tarnished by the MCC value of the yellow class but very well supported by the negative values of the orange and red class. This again provides

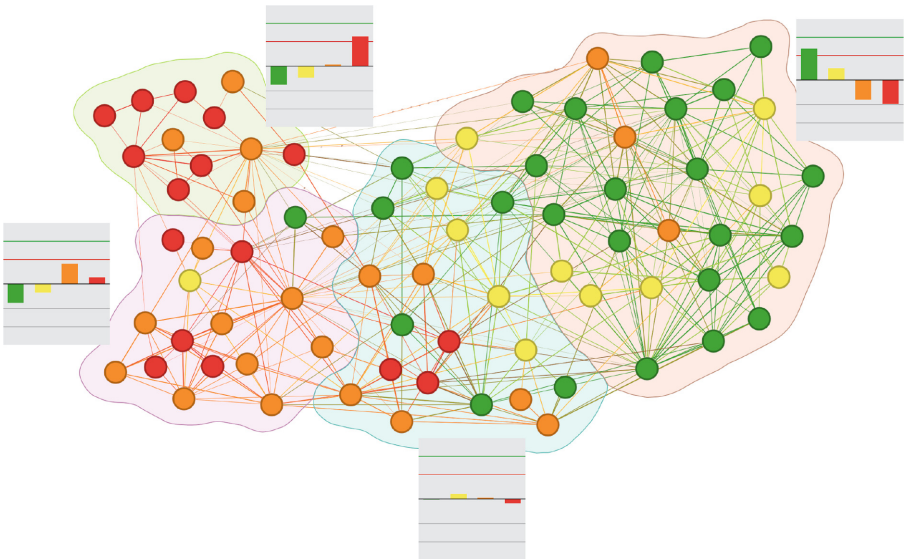


Fig. 6. A PSN with Class Dominancy Profiles.

more easily understandable information than the original evaluation ($P = 0.89$, $R = 0.68$. $F_1 = 0.77$).

In the updated evaluation of the network with our new method, it is apparent how this new approach provides a more understandable, explainable, and swift evaluation of a PSN compared to the previously employed techniques.

6 Conclusions

We presented an objective method of evaluating cluster class relationships in Patient Similarity Networks. Using the Matthews Correlation Coefficient to measure the dominancy of each class, we argued that prevalence does not necessarily mean importance and that the clinician's interpretation should be based on the dominant class of a cluster regarding the distribution of all classes in all clusters. In a network constructed from a real-world clinical dataset, we demonstrated how the visualization of the importance of individual classes (categorical labels, e.g., diagnoses or disease severity) is easily understandable and explainable using the dominancy profiles. We further demonstrated how the method helps improve the evaluation in cases where the class importance does not agree with the clinicians' intuition for imbalanced datasets.

We would like to note that our method is applicable to any multi-class, multi-cluster situation and is not necessarily limited to patient similarity networks or similarity networks in general. However, it is important for us to combine the visualization of datasets with clusters and classes and with dominance profiles, which is not easily solved without the use of networks.

Funding Information. This work was supported by SGS, VSB-Technical University Ostrava, under grant no. SP2024/007 and by MH CZ - DRO (FNOl, 00098892).

References

1. Ballabio, D., Grisoni, F., Todeschini, R.: Multivariate comparison of classification performance measures. Chemom. Intell. Lab. Syst. **174**, 33–44 (2018)
2. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Statist. Mech. Theory Exp. **2008**(10), P10,008 (2008)
3. Boughorbel, S., Jarray, F., El-Anbari, M.: Optimal classifier for imbalanced data using Matthews correlation coefficient metric. PloS one **12**(6), e0177,678 (2017)
4. Brown, J.: Classifiers and their metrics quantified. Mol. Inform. **37**(1-2), 1700,127 (2018)
5. Bulavas, V., Marcinkevičius, V., Rumiński, J.: Study of multi-class classification algorithms' performance on highly imbalanced network intrusion datasets. Informatica **32**(3), 441–475 (2021)
6. Chicco, D., Jurman, G.: The advantages of the Matthews correlation coefficient (MCC) over f1 score and accuracy in binary classification evaluation. BMC Genom. **21**, 1–13 (2020)

7. Chicco, D., Jurman, G.: The Matthews correlation coefficient (mcc) should replace the ROC AUC as the standard metric for assessing binary classification. BioData Mining **16**(1), 4 (2023)
8. Gorodkin, J.: Comparing two k-category assignments by a k-category correlation coefficient. Comput. Biol. Chem. **28**(5–6), 367–374 (2004)
9. Grandini, M., Bagli, E., Visani, G.: Metrics for multi-class classification: an overview. arXiv preprint arXiv:2008.05756 (2020)
10. Janca, O., Ochodkova, E., Kriegova, E., Horak, P., Skacelova, M., Kudelka, M.: Real-world data in rheumatoid arthritis: patient similarity networks as a tool for clinical evaluation of disease activity. Appl. Netw. Sci. **8**(1), 57 (2023)
11. Kuo, K.M., Talley, P., Kao, Y., Huang, C.H.: A multi-class classification model for supporting the diagnosis of type ii diabetes mellitus. PeerJ **8**, e9920 (2020)
12. Matthews, B.W.: Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA)-Protein Structure **405**(2), 442–451 (1975)
13. Ochodkova, E., Zehnalova, S., Kudelka, M.: Graph construction based on local representativeness. In: Cao, Y., Chen, J. (eds.) Computing and Combinatorics, pp. 654–665. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-62389-4_54
14. Ochodkova, E., Zehnalova, S., Kudelka, M.: Graph construction based on local representativeness. In: Cao, Y., Chen, J. (eds.) COCOON 2017. LNCS, vol. 10392, pp. 654–665. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62389-4_54
15. Opitz, J.: From bias and prevalence to macro f1, Kappa, and MCC: a structured overview of metrics for multi-class evaluation. Heidelberg University (2022)
16. Sova, M., et al.: Network analysis for uncovering the relationship between host response and clinical factors to virus pathogen: lessons from SARS-COV-2. Viruses **14**(11), 2422 (2022)
17. Trajerová, M., Kriegová, E., Mikulková, Z., Savara, J., Kudelka, M., Gallo, J.: Knee osteoarthritis phenotypes based on synovial fluid immune cells correlate with clinical outcome trajectories. Osteoarthr. Cartil. **30**(12), 1583–1592 (2022)

Structural Asymmetries in PPI Networks as a Tool to Improve the Detection of Protein Complexes

Lukas Papik[1](✉), Eliska Ochodkova[1], Eva Kriegova[2], and Milos Kudelka[1]

[1] VSB - Technical University of Ostrava, Ostrava, Czech Republic
lukas.papik@vsb.cz

[2] Palacky University Olomouc and University Hospital, Olomouc, Czech Republic

Abstract. A protein complex is a group of two or more proteins that interact to perform specific biological functions. Protein complexes are essential for cellular processes, and their understanding is still limited. Therefore, the detection of protein complexes is an important topic in bioinformatics, often using protein-protein interaction (PPI) networks. There are many approaches that can detect tightly connected subgroups of proteins in PPI networks that correspond to known reference protein complexes or that may be the subject of further investigation based on a biological assessment of their significance. Accurate detection is still problematic due to the complexity of PPI networks and the dynamic and asymmetric nature of protein interactions. However, PPI networks usually do not capture such properties. The novel approach presented in this paper exploits the asymmetric relationship between pairs of proteins based on analyzing the network structures in their neighborhood. We show that a simple *mDepStar* method exploiting this structural asymmetry outperforms most of the six state-of-the-art detection methods using only the topological properties of the network. Comparisons were performed on two well-known PPI networks and a newly constructed BioGRID network with a new SGD24 reference set for *Saccharomyces cerevisiae*. In addition, we introduce a new MR-score inspired by the F-measure that incorporates the most commonly used Maximum Matching Ratio.

Keywords: protein · complex · PPI network · structural asymmetry

1 Introduction

Proteins play a key role in mediating metabolic processes, signaling pathways, signaling cascades, and more. Proteins usually do not function alone but often interact with other proteins. Such interacting groups of proteins that perform specific functions are called protein complexes, which are structures composed of several proteins that function together in different biological processes; many computational methods have been proposed to predict protein complexes [26].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 27–38, 2025.
https://doi.org/10.1007/978-3-031-82439-5_3

Studying protein complexes helps to understand the role of proteins in the organism or the differences between organisms, as the functions of many proteins are still unknown. Protein interactions can be captured in so-called PPI networks, where proteins are represented as nodes and their interactions as edges. Identifying protein interactions needed to construct PPI networks has been made possible by high-throughput methods, such as affinity purification and mass spectrometry [7].

To understand not only how the proteins are organized into functional units but also the higher-level cellular organization, it is essential to be able to predict protein complexes in PPI networks [6,17]. In the past, the starting point for correct prediction was the assumption that protein complexes in PPI networks tend to be larger and more densely connected [1,3,12]. Therefore, the methods used (and are still being used) features such as high clustering coefficient, strong community structure, etc. However, it has gradually become clear that protein complexes can be small and sparse [1] as well as large and dense and can overlap significantly [15]. Another problem is that not all interactions are known, and PPI networks do not provide reliable information about the connectivity between proteins, leading to false positive and false negative predictions [20]. In addition, even known complexes may not be determined correctly. Therefore, inaccuracies and duplicate predictions of the same known protein complex are to be expected when evaluating the quality of methods. The prediction here means (1) the detection of known (reference) protein complexes and (2) the detection of a group of proteins that do not match the reference complexes but may be subject to further biological evaluation.

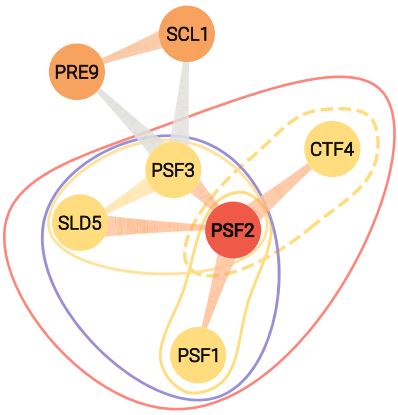


Fig. 1. Essential protein complex GINS in Saccharomyces cerevisiae. In the group of seven proteins, the GINS reference complex is outlined in blue, and the other GINS variant is outlined in red [8]. The red one is predicted by the *mDepStar* method together with the other predictions outlined in yellow. The edges of the network represent structurally asymmetric relationships between proteins.

An example of the relationship between the predicted and reference protein complexes (prediction and reference for short) is shown in Fig. 1 for a subnetwork of the KroganCore PPI network and CYC reference set with a total of seven proteins. The blue line indicates the reference complex with four proteins, and the red and yellow lines indicate the complexes predicted by the novel *mDepStar* (Mutually Dependent Star) method described below. The dashed line indicates one predicted complex that does not fit any reference complex; the other three fit the blue reference complex. The two orange nodes at the top of the figure are connected to the rest of the network.

Here, we propose a method mDepStar based solely on topological information in a PPI network that allows us to asymmetrically quantify the dependency relationship between pairs of interacting proteins in a PPI network. This quantification shows that *mutually dependent proteins* in our benchmark networks exhibit high semantic similarity of cellular components, which may indicate that these protein pairs may be part of the same protein complex [24]. We are able to predict protein complexes that well satisfy recently known requirements. Compared with six state-of-the-art methods, we show that our predicted complexes achieve the highest scores among almost all other methods in terms of matching reference complexes and pairwise Gene Ontology semantic similarities.

2 PPI Networks and Reference Sets

For experiments with *Saccharomyces cerevisiae* (baker's yeast), we selected three PPI benchmark networks and two reference sets. The two weighted PPI networks Collins [5] and KroganCore [10] are older and widely used. The third is the newly constructed BioGRID network. Because small network components may not provide enough reliable structural information needed for complex prediction, we performed experiments on giant components of these networks. Their properties are listed in Table 1. For each network, the number of nodes (n) and edges (m), the minimum, maximum and average degree, the average and maximum distance, the average clustering coefficient (CC), the assortativity (r), and the Louvain modularity (Q) are given.

For the Collins and KroganCore networks, we selected the frequently used older reference set CYC2008 (CYC for short) [19] with 408 manually curated protein complexes from the literature up to 2008. For the BioGRID network, we prepared a new reference set SGD24. Both reference sets were preprocessed by [15]: (1) removing proteins from each reference that were not present in the network and then (2) removing those references that did not contain at least half of the proteins from the original size. The counts and sizes for the preprocessed reference sets for each network are given in the last row in Table 2.

BioGRID Network has been constructed from the BioGRID [18] database version 4.4.231, using only physical protein interactions. Although the interactions are weighted, we constructed the network as unweighted due to the different weighting systems in the different experimental methods.

SGD24 Reference Set has been prepared to reflect the latest knowledge about yeast protein complexes according to the [15] approach using Gene Ontology (GO); GO is a system that provides a unified vocabulary for describing the functions of genes and gene products [2]. Individual terms with a unique identifier in GO represent specific molecular functions, biological processes, and cellular components; each GO term can be linked to other terms by relationships such as "is a" or "part of." The terms can then be arranged in a directed acyclic graph, where more specific terms inherit the properties of more general terms. In the construction of SGD24, a GO term with an "is a" relationship was selected for each cellular component of the protein-containing complex (PCC, GO:0032991). Each such PCC GO term then defines a single reference complex with a minimum size of two proteins. The reference set was created on 2024-08-04 using the QuickGO tool [4], and the total number of reference complexes is 506.

Table 1. Properties of PPI networks (giant components).

network	weights	n	m	degree			distance		CC	r	Q
				min	max	avg	avg	max			
BioGRID	No	5503	159981	1	3530	58.143	2.138	5	0.370	−0.182	0.313
Collins	Yes	1004	8323	1	127	16.579	5.542	15	0.647	0.556	0.760
KroganCore	Yes	2559	7031	1	141	5.495	4.744	12	0.194	−0.020	0.704

3 Structural Dependency as an Asymmetric Relationship

The structural dependency [11] is asymmetric and calculated between nodes, which are connected by an edge. If we had two nodes A and B connected by an edge, the structural dependency combines the neighborhood of those two nodes and weights $w(A, B)$ between them and to the neighbors of those two nodes connected by the edge. We take into account how many common neighbors and how strongly they are connected to, and on the other hand, how many and how strongly each of them is connected separately. We then define the dependency $D(A, B)$ of node A on node B as follows:

$$D(A,B) = \frac{w(A,B) + \sum_{X_i \in \Gamma(A,B)} W(A,B,X_i)}{\sum_{X_i \in N(A)} w(A,X_i)} \tag{1}$$

The weight W of the relationship between two nodes A, B given by their common neighbor X is half their harmonic mean

$$W(A,B,X) = W(B,A,X) = \frac{w(A,X) \cdot w(B,X)}{w(A,X) + w(B,X)} \tag{2}$$

$\Gamma(A, B)$ is the set of common neighbors of nodes A, B, and $N(A)$ is the neighborhood (set of all neighbors) of node A. A dependency defined in this

way is asymmetric; although the values of the numerators are the same in both directions of the dependency, the values of the denominators may be different. It is important to note here that the asymmetry holds even in unweighted networks, and in the computation, for each common neighbor X_i, the $W(A, B, X_i)$ is 0.5. The structural dependency is between $(0, 1]$.

Mutual Dependency. The question now is how to use dependency to determine that an asymmetric relationship between neighboring proteins could correspond to an interaction that is potentially part of a protein complex. We have two pieces of information in the PPI network to determine this. The first is the edge (interaction) weight, but this is only available in weighted networks and usually cannot be fully relied upon. The second piece of information is the well-known fact that the interaction is strengthened by interactions in the neighborhood of the pair of interacting proteins [9]. The dependency defined above takes both into account but can also be applied to unweighted networks.

Our experiments have shown that only pairs of proteins should be considered as components of protein complexes if (1) none of their (asymmetric) dependencies is too low and (2) at least one of them is high enough. Pairs of proteins that satisfy both conditions we call *mutually dependent*. A well-working heuristic for the first condition is to set the dependency threshold τ to half the weighted average of the average dependencies of neighboring proteins over the entire network; the weight for the average dependency in a pair of proteins is the lower of the two dependencies:

$$\tau = \frac{1}{2} \cdot \frac{\sum_{e \in Edges} min(D1, D2) \cdot avg(D1, D2)}{\sum_{e \in Edges} min(D1, D2)} \tag{3}$$

The second condition is set so that at least one of the two dependencies is above the $2 \cdot \tau$ threshold. Heuristics for determining both thresholds for both conditions have been tested on other PPI networks, including other organisms; a detailed presentation of all experiments is beyond the scope of this paper.

4 mDepStar Method

The name mDepStar stands for Mutually Dependent Star, and it consists of three simple steps[1]: (1) the dependencies of all adjacent node pairs (edges) are computed, and the dependency threshold τ is set; (2) each node is considered as a seed and together with all its mutually dependent neighbors as a predicted complex; (3) duplicate predictions are continuously removed. The time complexity of the algorithm is based on computing the dependencies for all pairs of adjacent nodes and is $O(mk^2)$, where m is the number of edges and k is the average degree of the network. The computation is trivially parallelizable.

In Fig. 1, the weighted KroganCore subnetwork shows the four complexes predicted by the mDepStar method and the GINS reference complex from the

[1] Python implementation and experimental data: https://github.com/l-pa/mDepStar.

CYC reference set outlined in blue. The red protein PSF2 was used as seed and forms, with four other yellow proteins, another variant of the GINS complex outlined in red [8], which matches the blue variant. Three other predictions with different seeds are outlined in yellow, of which the dashed one does not fit with the blue reference GINS complex and is the only false positive. Among the three yellow proteins on the left, the edge (interaction) between the SLD5 and PSF1 proteins is missing in the KroganCore network but is part of the BioGRID network. If this edge were present, the two pairs of true positive yellow complexes outlined in yellow would merge into the blue reference variant GINS complex. Asymmetric edge strengths reflect asymmetric mutual dependencies; for example, the two yellow PSF1 and CTF4 proteins on the right depend more on the red protein than the red protein depends on them, while the opposite is true for the other two yellow SLD5 and PSF3 proteins. The gray edges do not satisfy at least one of the mutual dependency conditions.

5 Evaluation Measures

There are many measures for assessing the match between predicted and reference complexes, a complete list of which is beyond the scope of this paper (see, e.g., [26]). The key measure here is the *Overlap Score* for the match between a prediction-reference pair, which is further applied in three binary-based and three OS-based measures. Semantic similarity GO is used for the biological evaluation of prediction quality. Efficient methods for predicting protein complexes should have high values in most criteria.

The Overlap Score (OS) between a pair of protein sets C_i and C_j is defined as [15,23]:

$$OS(C_i, C_j) = \frac{|C_i \cap C_j|^2}{|C_i||C_j|}. \tag{4}$$

For further use of OS, it is necessary to set a threshold that defines the binary match between the prediction and the reference. Then, a prediction is true positive if OS is above this threshold; in this case, we say that the prediction matches the reference and vice versa. We set the threshold to 0.25 for further evaluation [15].

Binary-Based Measures do not use the calculated value of the overlap score, but only the binary match between predictions and references ($OS > 0.25$). Let us further denote the set of predictions $Pred$ and references Ref, predictions that match at least one reference $Pred^+$, and references that match at least one prediction Ref^+. Then, the binary-based measures precision $Prec$, recall Rec, and $F\text{-}measure$ can be defined:

$$Prec = \frac{|Pred^+|}{|Pred|} \tag{5}$$

$$Rec = \frac{|Ref^{+}|}{|Ref|} \tag{6}$$

$$F\text{-}measure = \frac{2 \cdot Prec \cdot Rec}{Prec + Rec} \tag{7}$$

The F-*measure* balances the proportions of matching predictions and matching references. It does not consider the match's quality, but this is balanced by a well-understandable interpretation of all three measures.

OS**-Based Measures** typically work with the quality of the match (OS) for true positive predictions and corresponding references. The crucial measure is the Maximum Matching Ratio (MMR), which is computed using a bipartite graph with nodes representing predictions and references and edges between them with weights represented by OS. The MMR is then defined as the maximum one-to-one match between predictions and references divided by the total number of references. Maximizing the one-to-one match can be interpreted as an analogy or weighted version of recall. Next, we newly define a similar analogy for precision PMR, which allows us to apply the same idea as for the F-*measure*, namely to define a new measure MR-*score* that balances the two OS-based measures.

Predictive Matching Ratio (PMR**)** is a weighted version of precision; it sums the maximum OS for all positive predictions and divides them by the number of all predictions.

$$PMR = \frac{\sum_{P_i \in Pred^{+}} \max_{R_j \in Ref} OS(P_i, R_j)}{|Pred|} \tag{8}$$

Matching Ratio Score (MR -score**)** inspired by F-*measure*, is the harmonic mean of PMR and MMR and is its weighted version.

$$MR\text{-}score = \frac{2 \cdot MMR \cdot PMR}{MMR + PMR} \tag{9}$$

Figure 2 shows the experimental binary-based and OS-based results with the three networks described below. As will be shown in the experimental section, comparing binary-based and OS-based results can provide interesting information when interpreting the performance of individual methods.

5.1 GO Similarity Measure

Gene Ontology (GO) term similarity measurement is used, for example, to assess the functions of genes, to compare gene functions between different organisms, or to annotate newly discovered genes functionally. For biological evaluation of predicted complexes, term similarity in protein-protein interactions is important. Approaches to measure term similarity in GO can be broadly classified into structural, informational, and combined ones. We used Wang's [22] method, which considers both the structure of the ontology and its information content.

The similarity between two terms is estimated by aggregating the information content of all direct and indirect children of these terms in the ontology. Since we use reference sets based on a protein-containing complex, we restrict the similarity measure to the cellular component domain.

In our experiments, we used `mgeneSim()` from the `GOSemSim` package in R [25] with `measure=''Wang''` and `combine=''BMA''` parameters to measure the GO similarity of pairs of proteins. We then averaged these similarities across the individual predicted complexes. The results of the GO similarity measurements for the methods compared are shown in Fig. 3.

6 Experiment: Method Performance Comparison

We compared the mDepStar method with six recent methods that also use only topological properties of networks without incorporating biological information. For all six methods, we used original implementations with default parameters recommended by the authors. We set the minimum prediction size to 3 (reference size to 2), as is common in other comparative studies. Details of the predictions of each method, listed below, are summarized in Table 2.

BOPS Segments a network into sub-networks that are analyzed for coherence, and selected subsets are identified as complexes. [13].

ClusterOne Initiates clustering by selecting the highest-degree nodes and expanding them through a greedy process. Clusters are merged based on overlap then filtered. [15].

DPCMNE Compresses the network into multi-level subnetworks, computes embeddings using DeepWalk, and reconstructs a weighted network from which protein complexes are detected using core-attachment clustering. [14].

DPC-NADPIN Nodes with local clustering coefficients above a threshold form initial complexes, which are expanded based on affinity scores among neighboring nodes [21].

EWCA Reduces false positive by recalculating weights using the Jaccard coefficient. Nodes with similar neighbors above a threshold are considered cores, which are expanded by attaching compatible neighbors [23].

GCC-WCC Iteratively selects nodes with the highest clustering coefficient and their neighbors, forming clusters until no connected components remain [16].

Table 2 shows for each method and network-references combination the number of predictions, their maximum and average size, supplemented by quartiles giving more precise information about their size; the values can be compared with the references in the last row of the table. The DPC-NADPIN and EWCA methods have the largest predictions (with an extreme maximum of 3571 and an average of 60.5 for the BioGRID network), while the BOPS and GCC-WCC methods have the smallest predictions (with a low average of 3.9). The mDepStar method does not differ from most of the other methods (except for the high number of predictions for the BioGRID network).

Table 2. Predicted complexes for the compared methods. The last row shows the reference complexes after preprocessing for each benchmark network. The highest values for each column that are furthest from the reference values are highlighted.

method	BioGRID/SGD24						Collins/CYC						KroganCore/CYC					
	#	max	avg	Q1	Q2	Q3	#	max	avg	Q1	Q2	Q3	#	max	avg	Q1	Q2	Q3
mDepStar	2212	218	22.7	5	11	28	453	55	14.4	5	12	21	539	20	5.7	3	4	7
BOPS	565	16	3.6	3	3	4	**803**	19	5.6	4	5	7	681	18	4.6	3	4	5
ClusterOne	477	101	8.5	4	6	8	111	104	10.8	4	7	10	509	30	6.0	4	5	6
DPCMNE	952	138	15.3	4	7	16	357	89	18.9	7	15	25	662	24	6.3	3	5	7
DPC-NADPIN	**4107**	351	41.1	14	25	51	700	**129**	22.5	**8**	17	28	**714**	**38**	**19.7**	**5**	**8**	**13**
EWCA	1553	**3571**	**60.5**	**15**	**32**	**68**	549	109	**24.9**	**8**	**19**	**33**	665	33	8.3	4	6	12
GCC-WCC	611	71	5.7	3	4	6	116	36	6.3	3	4	7	291	14	3.9	3	3	4
references	485	131	6.3	2	4	7	116	62	7.3	2	4	10	272	29	4.5	2	3	5

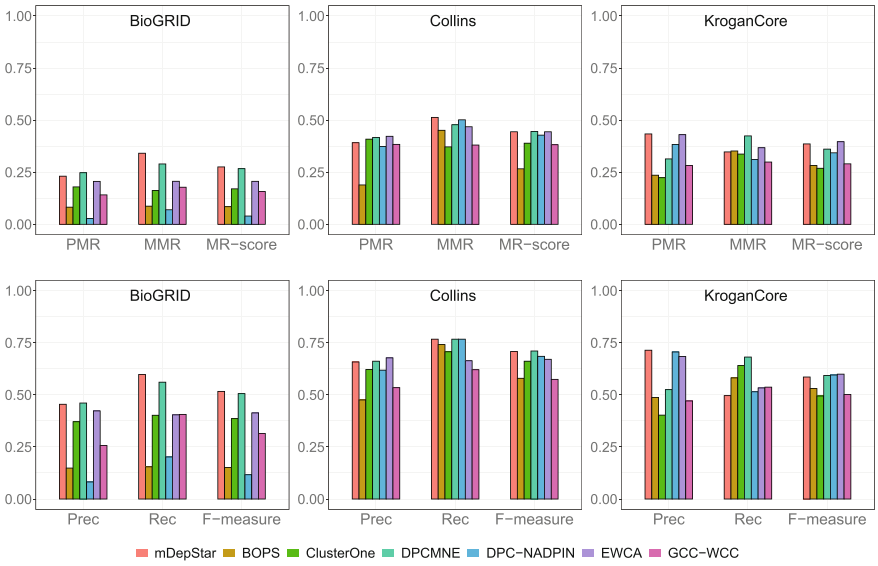


Fig. 2. Bar plots of *OS*-based and binary-based measures across methods.

The bar plots in Fig. 2 represent the individual structural measures for prediction and reference matching; the upper plots are *OS*-based, and the lower ones are binary-based measures. Only the mDepStar and DPCMNE methods have high *MR-score* and *F-measure* for all networks; EWCA has a slightly lower value in BioGRID. All other methods do not reach such high values, especially BOPS and DPC-NADPIN have very low values for the BioGRID network. For *PMR*, *MMR*, *Prec*, *Rec*, again, the methods mDepStar (except *Rec* for the KroganCore network) and DPCMNE (except *Prec* for the same network) have high values; EWCA is again only slightly worse. The difference between the *OS*-based and binary-based measures is most obvious for the BOPS method in the Collins network; the relatively high binary-based values do not correspond to the

lower OS-based values. The reason is that the BOPS method has a less accurate matching of predictions and references in the Collins network than the other methods. Another example is that the low Rec for mDepStar in the KroganCore network (lower proportion of correctly determined references) is compensated by the higher matching accuracy and, thus, higher MMR.

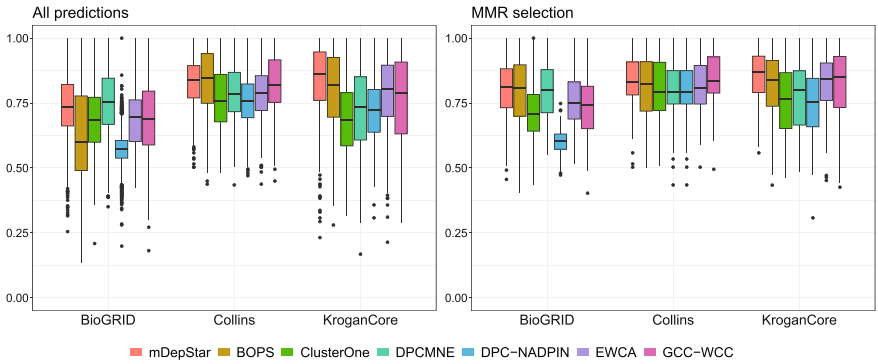


Fig. 3. Boxplots of average pairwise GO similarities for all (left) and best (right) predicted protein complexes across seven methods compared.

The boxplots in Fig. 3 show the average pairwise GO term similarities in the predicted complexes (left). On the right, the boxplots are only for the predictions that have the maximum match with unique references (i.e., those found when maximizing the match in the MMR calculation). The figures show that when we consider only the best predictions, the results of the methods improve and balance, and almost all predictions have fewer outliers and average pairwise GO similarities above 0.5. The improvement in BOPS results in the BioGRID network is apparently caused by many predictions with low GO similarity. There is no significant improvement in mDepStar (and some others), showing that most of the predicted complexes have relatively high GO similarities. The excellent results of the mDepStar method are confirmed by high medians (first or second highest in all cases); narrow interquartile ranges show that the results are reliable and consistent. In contrast, DPMNE and EWCA have some worse results.

We can also see in Figs. 2, 3 the relationship between the performance of the methods and the properties of the networks in Table 1. The results for the Collins network with a high average clustering coefficient, assortativity, and modularity are almost comparable. In contrast, for the BioGRID network, where all these values are much lower, the performance of the methods compared is very different. However, regardless of the properties of the networks, when binary-based, OS-based, and biological GO similarity results are considered together, the mDepStar method, despite its simplicity, achieves the best overall results.

7 Conclusion

As our experiments have shown, mutual asymmetric dependency seems to be a promising structural approach to identifying such interacting pairs of proteins that may be part of protein complexes. Although protein interactions are known to be asymmetric, this property is not exploited in recent detection methods. However, using structural asymmetry can highlight its importance and provide an advantage in detecting protein complexes.

The weakness of the mDepStar method is that the predicted complex consists only of a seed and its selected neighbors. Therefore, the maximum network distance between the proteins of a complex is 2. However, in PPI networks, distances within complexes can sometimes be slightly greater. Refining the mDepStar method to account for greater protein distances is the first goal of further research. For large-scale networks, the second goal is to analyze dependencies around seed nodes to allow individual estimation of mutual dependency thresholds without having to analyze the entire network.

Funding Information. This work was supported by SGS, VŠB-Technical University Ostrava, under grant no. SP2024/007 and by MH CZ - DRO (FNOl, 00098892).

References

1. Adamcsek, B., Palla, G., Farkas, I.J., Derényi, I., Vicsek, T.: CFinder: locating cliques and overlapping modules in biological networks. Bioinformatics **22**(8), 1021–1023 (2006)
2. Aleksander, S.A., et al.: The gene ontology knowledgebase in 2023. Genetics **224**(1), iyad031 (2023)
3. Bader, G.D., Hogue, C.W.: An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinform. **4**, 1–27 (2003)
4. Binns, D., Dimmer, E., Huntley, R., Barrell, D., O'donovan, C., Apweiler, R.: Quickgo: a web-based tool for gene ontology searching. Bioinformatics **25**(22), 3045–3046 (2009)
5. Collins, S.R., et al.: Toward a comprehensive atlas of the physical interactome of saccharomyces cerevisiae. Mol. Cell. Prot. **6**(3), 439–450 (2007)
6. Gavin, A.C., et al.: Functional organization of the yeast proteome by systematic analysis of protein complexes. Nature **415**(6868), 141–147 (2002)
7. Gnanasekaran, P., Pappu, H.R.: Affinity purification-mass spectroscopy (AP-MS) and co-immunoprecipitation (Co-IP) technique to study protein–protein interactions. In: Mukhtar, S. (eds.) Protein-Protein Interactions. Methods in Molecular Biology, vol. 2690, pp. 81–85. Humana, New York, NY (2023). https://doi.org/10.1007/978-1-0716-3327-4_7
8. Hart, G.T., Lee, I., Marcotte, E.M.: A high-accuracy consensus map of yeast protein complexes reveals modular nature of gene essentiality. BMC Bioinform. **8**, 1–11 (2007)
9. Kovács, I.A., et al.: Network-based prediction of protein interactions. Nat. Commun. **10**(1), 1240 (2019)
10. Krogan, N.J., et al.: Global landscape of protein complexes in the yeast saccharomyces cerevisiae. Nature **440**(7084), 637–643 (2006)

11. Kudelka, M., Ochodkova, E., Zehnalova, S., Plesnik, J.: Ego-zones: non-symmetric dependencies reveal network groups with large and dense overlaps. Appl. Netw. Sci. **4**(1), 1–49 (2019)
12. Lubovac, Z., Gamalielsson, J., Olsson, B.: Combining functional and topological properties to identify core modules in protein interaction networks. Proteins Struct. Funct. Bioinform. **64**(4), 948–959 (2006)
13. Lyu, J., Yao, Z., Liang, B., Liu, Y., Zhang, Y.: Small protein complex prediction algorithm based on protein-protein interaction network segmentation. BMC Bioinform. **23**(1), 405 (2022)
14. Meng, X., Xiang, J., Zheng, R., Wu, F.X., Li, M.: DPCMNE: detecting protein complexes from protein-protein interaction networks via multi-level network embedding. IEEE/ACM Trans. Comput. Biol. Bioinf. **19**(3), 1592–1602 (2021)
15. Nepusz, T., Yu, H., Paccanaro, A.: Detecting overlapping protein complexes in protein-protein interaction networks. Nat. Methods **9**(5), 471–472 (2012)
16. Omranian, S., Angeleska, A., Nikoloski, Z.: Efficient and accurate identification of protein complexes from protein-protein interaction networks based on the clustering coefficient. Comput. Struct. Biotechnol. J. **19**, 5255–5263 (2021)
17. Omranian, S., Nikoloski, Z., Grimm, D.G.: Computational identification of protein complexes from network interactions: present state, challenges, and the way forward. Comput. Struct. Biotechnol. J. **20**, 2699–2712 (2022)
18. Oughtred, R., et al.: The biogrid interaction database: 2019 update. Nucleic Acids Res. **47**(D1), D529–D541 (2019)
19. Pu, S., Wong, J., Turner, B., Cho, E., Wodak, S.J.: Up-to-date catalogues of yeast protein complexes. Nucleic Acids Res. **37**(3), 825–831 (2009)
20. Safari-Alighiarloo, N., Taghizadeh, M., Rezaei-Tavirani, M., Goliaei, B., Peyvandi, A.A.: Protein-protein interaction networks (PPI) and complex diseases. Gastroenterol. Hepatol. Bed Bench **7**(1), 17 (2014)
21. Shen, X., et al.: Neighbor affinity based algorithm for discovering temporal protein complex from dynamic PPI network. Methods **110**, 90–96 (2016)
22. Wang, J.Z., Du, Z., Payattakool, R., Yu, P.S., Chen, C.F.: A new method to measure the semantic similarity of go terms. Bioinformatics **23**(10), 1274–1281 (2007)
23. Wang, R., Liu, G., Wang, C.: Identifying protein complexes based on an edge weight algorithm and core-attachment structure. BMC Bioinform. **20**, 1–20 (2019)
24. Wang, S., Wu, F.: Detecting overlapping protein complexes in PPI networks based on robustness. Prot. Sci. **11**, 1–8 (2013)
25. Yu, G., Li, F., Qin, Y., Bo, X., Wu, Y., Wang, S.: Gosemsim: an R package for measuring semantic similarity among go terms and gene products. Bioinformatics **26**(7), 976–978 (2010)
26. Zahiri, J., et al.: Protein complex prediction: a survey. Genomics **112**(1), 174–183 (2020)

A Physiologically-Grounded, Computationally-Efficient Model of the Human Hippocampus

Niccolò Calcini(✉), Vitalii Kyzym, and Renaud B. Jolivet

Maastricht Centre for Systems Biology (MaCSBio), Maastricht University, Maastricht, The Netherlands
n.calcini@neurophysiology.nl

Abstract. The hippocampus is a central structure of the vertebrate brain, and plays a key role in the formation and processing of memories. Comprised of several highly interconnected sub-regions, it is characterized by a complex network architecture involving both feed-forward and feedback loops. In the past, various computational models of the hippocampus have been proposed. However, these were either lacking in physiological details, or required significant computational resources. With this work, we propose a novel human hippocampal model, which is both solidly grounded in physiological details, and has low computational requirements, allowing for the investigation of various physiological or pathological conditions. Here, we describe the model in detail, and demonstrate how its architecture supports firing rates typical of a resting hippocampus, and the spontaneous generation of characteristic hippocampal theta oscillations.

Keywords: Hippocampus · Simulation · Oscillations · Complex Systems

1 Introduction

The hippocampus is a critical brain structure located in the medial temporal lobe, which plays an essential role in memory formation, spatial navigation, and emotions regulation [1]. Understanding its complex structure and function is fundamental to advancing our knowledge of cognitive processes, memory formation, and neurological disorders. Given the involvement of the hippocampus in both declarative memory and spatial processing [2], simulating its activity offers a unique opportunity to deepen our understanding of its functional mechanisms. Moreover, its distinct cellular architecture, organized into layers and sub-regions such as the entorhinal cortex (EC), dentate gyrus (DG), and cornu ammonis (CA1-2–3), makes it an intriguing target for computational models that seek to unravel its network dynamics [3]. These sub-regions, while anatomically distinct,

N. Calcini and V. Kyzym—Contributed equally.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 39–50, 2025.
https://doi.org/10.1007/978-3-031-82439-5_4

are functionally interconnected, allowing for complex information processing and propagation across the whole hippocampal formation [4].

The primary function of the hippocampus is thought to be the encoding, consolidation, and retrieval of episodic memories [5]. Its structure, in particular, plays a key role in spatial memory, as demonstrated in key studies that identified place cells, neurons which activate in response to specific spatial locations in a certain environment [6]. This spatial coding is essential for navigation and orientation, allowing organisms to map and recall locations in their environment. Additionally, the hippocampus is involved in pattern separation and completion, that is the ability to, respectively, distinguish between similar stimuli, or to reconstruct a complete memory from partial inputs [7].

Characterized by high degrees of connectivity, different pathways, and a network architecture involving both feed-forward and feedback loops [8], the hippocampus represents one of the most complex regions of the brain. Dynamic processes contribute to the formation of neural assemblies, which allow for the encoding and retrieval of memories [9], and the recursive feedback that characterize this system ensures that the hippocampus operates in a highly non-linear manner, a hallmark of complex adaptive systems [10,11].

By implementing detailed simulations of the hippocampus, it is thus possible to explore the computational principles that underlie its function with greater precision and control, enabling the study of how these neural circuits contribute to cognition, and potentially revealing insights into neurodegenerative conditions such as Alzheimer's disease and epilepsy, where hippocampal dysfunction is often a prominent feature [12]. Finally, such a line of study might also serve as a testing ground for theories of emergent properties in neuronal networks.

Previous computational models of the hippocampus have primarily focused either on simplified, macroscopic representations, or on addressing small-scale circuits and single sub-areas [13], or abstracting the system at a high level [14,15]. However, there remains a gap in the literature concerning detailed simulations that integrate both the microcircuit dynamics and the macroscopic, large-scale interactions within the hippocampus, that is computationally accessible, physiologically realistic, and exhibiting the emerging functional properties of this area. Our model offers to bridge that gap, providing a platform to investigate the behavior of hippocampal circuits under varying conditions.

2 Methods

2.1 Geometry and Topology

The model divides the hippocampus in five sub-regions: entorhinal cortex (EC), subiculum (Sub), cornus ammonis 1 (CA1), cornus ammonis 3 (CA3), and dentate gyrus (DG). The geometry of the hippocampal model was constructed using rectangular blocks, with each block representing one of the five hippocampal subfields. To accurately position and proportion the blocks, histological data from the Big Brain project were utilized, consisting of serially section and stained

brain tissue at 40 μm^3 resolution [16]. Following a previously performed histological tracing of the hippocampal sub-fields in slide 376 of the left hippocampus [17], a block segmentation was obtained. Simple block geometry (Fig. 1) provides realistic proportions for each segment, and follows the hippocampal curvature. A depth of 5 mm was chosen for the model geometry as the coronal view of the hippocampus remains relatively unchanged within that range. As a result, each sub-field was modeled as a 3D block with dimension outlined in Table 1.

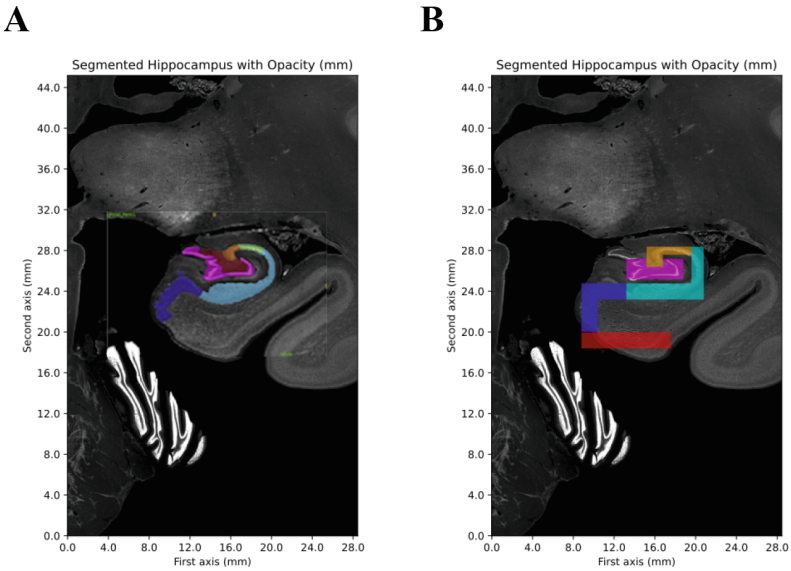


Fig. 1. A) The hippocampus segmentation as done by DeKraker *et al.* [17]. B) The matching reconstruction with rectangular volumes used in our model. The image shown is an adaptation from DeKraker *et al.*, under a Creative Commons license (used with the author's permission).

The model was simulated as a scaled down version of the hippocampus with 20000 simulated point neurons. Within each block, the neurons were assigned an excitatory-to-inhibitory ratio of 85% to 15%, in accordance with literature values for the hippocampus (see Table 2) [18,19]. To ensure that the scaled down model retained a correct neuron density, the model geometry was scaled accordingly. Specifically, the sides of each block were scaled by $\sqrt{\#\text{ of simulated neurons}/\#\text{ of expected neurons}}$ to maintain the desired neuron density in each region.

2.2 Network Simulations

The neural network simulations were performed using Brian2 [20], employing single-compartment models based on the Hodgkin-Huxley formalism. The neuron models used [21], were derived from earlier work on hippocampal neurons

Table 1. Neuronal densities, number of neurons, and number of simulated neurons in the scaled down model for each of the modelled hippocampal regions.

Region	Density (cells/mm^3)	Neurons (# expected)	(# simulated)
DG	87100	4,320,9600	7305
CA3	17548	477,7616	806
CA1	51902	4,280,1280	7232
Sub	21128	1,284,6080	2170
EC	20803	1,464,9216	2476

Table 2. Dimensions of the rectangular volumes used to represent each hippocampal region.

Region	Length (mm)	Width (mm)	Depth (mm)
EC	8.8	1.6	5.0
Sub (left)	1.6	3.2	5.0
Sub (right)	4.4	1.6	5.0
CA1 (left)	6.4	1.6	5.0
CA1 (right)	1.2	5.2	5.0
CA3 (right)	2.8	0.8	5.0
CA3 (left)	1.6	2.0	5.0
DG (right)	2.0	2.0	5.0
DG (middle)	1.6	1.2	5.0
DG (left)	2.0	2.0	5.0

[22]. The modelled neurons contain a sodium, potassium, slow non-inactivating potassium current, and a leak current. The synapses between neurons were taken from synaptic models of hippocampal AMPA and GABA-A synapses [23]. Neurons were connected according to a distance decay relationship modeled by a Gaussian distribution:

$$P(x, y, z) = A \exp\left(-\frac{x^2 + y^2 + z^2}{2\sigma^2}\right). \tag{1}$$

Inhibitory neurons followed a distance-dependent probability distribution with a standard deviation of 4.5 mm, a setting that matched experimental conditions for interneurons in the CA1 region. Experimental observations noted that the connection between CA1 basket cells and pyramidal neurons was approximately 30%, decreasing with distance from the soma with a peak at 54% for nearby neurons [24]. For this reason, we set $A = 0.54$ and $\sigma = 4.5$ mm for the i2e distance-dependent probability of connection. In contrast, excitatory neurons are known to have longer axons to transmit signals between distant hippocampal regions. Depending on cell type and region, axon length can vary, ranging anywhere from 4000 μm in the CA3 [25] up to 11415 μm in the DG region [26].

Based on this, we selected $\sigma = 7.5\,\text{mm}$ for excitatory neuron projections. A noisy current was separately injected into each neuron in the network to activate them. That current was modeled as an Ornstein-Uhlenbeck process given by:

$$\frac{dI_{\text{noise}}}{dt} = \frac{\mu_{\text{noise}} - I_{\text{noise}}}{\tau_{\text{noise}}} + \text{noise}, \tag{2}$$

with noise given by:

$$\text{noise} = \sigma_{\text{noise}} \sqrt{\frac{3}{\tau_{\text{noise}}}} \frac{\text{randn}()}{\sqrt{dt}}, \tag{3}$$

and with $\tau_{\text{noise}} = 2\,\text{ms}$. Excitatory neurons received noise with a mean between 60 and 140 pA, and a standard deviation between 10 and 70 pA, matching current levels commonly found to elicit action potential in patch clamp experiments [36], and typical input levels for a resting state hippocampus [37]. The inhibitory neurons were activated in the same way, but with a mean between 55 and 95 pA, and a standard deviation between 10 and 20 pA. This choice was made to simulate the fact that the activity of inhibitory neurons fluctuates less, and stabilizes network activity [38].

2.3 Intra-Regional Connectivity

Excitatory and inhibitory neurons within each region were connected based on experimentally determined probabilities for each neural pair. Electrophysiological recordings of pyramidal cells in the CA3 suggest that the probability of two cells being connected is around 2% [39], though some estimates reach as high as 15% [40]. Nevertheless, the excitatory-to-excitatory (e2e) connectivity in the CA3 is relatively low and is reflected by an 8.9% probability in our model. Likewise, the probability of an excitatory-to-inhibitory neuron connection (e2i) in the CA3 is estimated to be around 10% [41]. The corresponding probability in our model is 8.6%. The inhibitory-to-excitatory probability of connection (i2e) observed in the CA3 ranges from 30% for chandelier cells [42], to 60% for basket cells [43]). Since the model currently does not distinguish between different inhibitory neurons sub-types, we used the value 52.2%.

The connection probability between two pyramidal cells within the CA1 region is approximately 1 in 130 [44]. Nevertheless, to ensure consistent excitatory activity in the region, a low value of 3.7% was chosen for e2e connection probability in our model. The connection probability from CA1 pyramidal neurons to stratum pyramidale inhibitory interneurons (e2i) was determined to be 30% [24]. Furthermore, the connectivity of the basket-to-pyramidal pair in the CA1 was estimated at 30% [45], which is the value we used for the i2e value of our model. Lastly, there is recurrent inhibitory connectivity occurring between the interneurons of the CA1 at a high rate, up to 50% [46], approximated to 39.2% in our model. Within the subiculum, the e2e connectivity rate is estimated to be 4% [47], although for bursting cells it is reported to be around 7.4% [48]. A value of 6.7% was chosen for our model to approximate this range. Connection probabilities between pyramidal and inhibitory neurons range from

11–26%, and the i2e probability was generally higher, around 35–46% [47,48]. The connectivity used in our model adopted values within those ranges, as well as an additional i2i connection probability of 19.2% [48].

The e2e probability of connections in the EC, depends on their cell type: since we use a single excitatory type in our model, we used 13.5% of connection [49]. Furthermore, the inhibitory connections (e2i, i2e, i2i) were determined based on the output layer (layer II of the lateral EC) and subsequently utilized in our model as described in [31]. In the DG, the e2e probabilities are sparse and estimated to be around 5–10% [31,50]. For excitatory-inhibitory pairs (granule cell to PV+ interneurons) in the DG, connections were made with a mean probability of 11%, while i2e connectivity was observed at an average of 28.8% [51]. Therefore, our model incorporated a detailed connectivity map, pairing each neuron group in the region with an experimentally determined probability. The values describing the connectivity of each region are summarized in Table 3.

Table 3. Total probability of excitatory-excitatory connections (%) between each hippocampal subregion

	DG	CA3	CA1	Sub	EC
e2e	8.9	8.7	3.7	6.7	14
e2i	25.5	8.6	25.9	16.7	28.1
i2e	13.2	52.2	29.7	47.1	36.9
i2i	–	–	39.2	19.2	–

2.4 Inter-Regional Connectivity

Connectivity between adjacent regions in the hippocampus follows the previously mentioned distance-dependent connectivity rule. However, for non-adjacent regions, connections are assigned a uniform probability, independent of the distance between neurons. As the major input to the hippocampus, the perforant pathway (PP) projects signals from the EC to the DG, CA3 and CA1. It has been observed that the PP induces strong excitatory and inhibitory responses in each target area [27]. The projections produce equal excitability in all three regions [28], and contribute to approximately 30% of the evoked excitatory postsynaptic potential (EPSP) responses [29]. As a result, PP projections are assigned with a probability of connection of 25% in our model. Once the PP activity reaches the hippocampus, it propagates through the hippocampus from the DG → CA3 → CA1 → Sub → EC. The DG-to-CA3 projections are sparse [30], with an estimated connection probability of less than 10% [31]. In contrast, the CA3-to-CA1 pathway, known as the Schaffer collateral, exhibits dense excitatory and inhibitory connectivity. Pyramidal neurons in the CA3 can form connections with 50% of CA1 neurons [32], although some estimates suggest a connectivity

as high as 85% [33]. Our model adopts a more conservative estimate of 54%. Lastly, the Subiculum serves as the primary output region of the hippocampus and receives substantial input from the CA1, albeit with lower connectivity than the CA3-to-CA1 projections [34]. Due to the limited data on the Subicular inter-regional connection probabilities, our model estimates the values based on functional connectivity measurement in relation to other hippocampal regions [35]. The final inter-regional connectivity schematic is shown in Table 4.

Table 4. Probability matrix of connections between different hippocampal regions

From/To	EC	DG	CA3	CA1	Sub
EC	0	24.9	24.9	25.0	0
DG	0	0	10.0	0	0
CA3	0	0	0	54.0	0
CA1	25.0	0	0	0	24.3
Sub	14.0	0	0	0	0

3 Results

To characterise the model and demonstrate that its resulting neural activity is in line with what is known of its basic neurophysiological properties, we activated the network with a noise stimulus as described in the Methods section. In 15 s long simulations, we observed how the average firing rate of the excitatory neuron populations evolves for each sub-region (Fig. 2).

The average firing rate the excitatory neurons achieved by receiving the aforementioned sustained current injection, matched physiologically plausible values for a resting hippocampus [52,53]. We hypothesized that the observed initial high values in neural activity are due to a lag in the activation of the inhibitory neuron population, which, once recruited, brings the system to a steady-state. Further, we calculated the local field potential (LFP), showing that it similarly matched with what has been observed experimentally. Finally, we computed the power spectrum density (PSD) based on population firing rates, demonstrating how our model reproduces the spontaneous theta rhythm ($5 \sim 12$ Hz) that is a characteristic signature of hippocampal activity [52].

To deepen our characterisation of the model, we compared the anatomical connectivity between each sub-region, measured as the total probability of excitatory-excitatory connections, and the functional connectivity between the same sub-regions, measured as the transfer entropy between them (Fig. 3). Finally, to compare between the two connectivity matrices, we computed the cosine similarity between each of them, after normalising them, and taking into consideration only regions with a non-zero anatomical connectivity (Fig. 3). The results showed the presence of a non-trivial relation between the two measures, hinting at inherent non-linearities in the flow of neural activity.

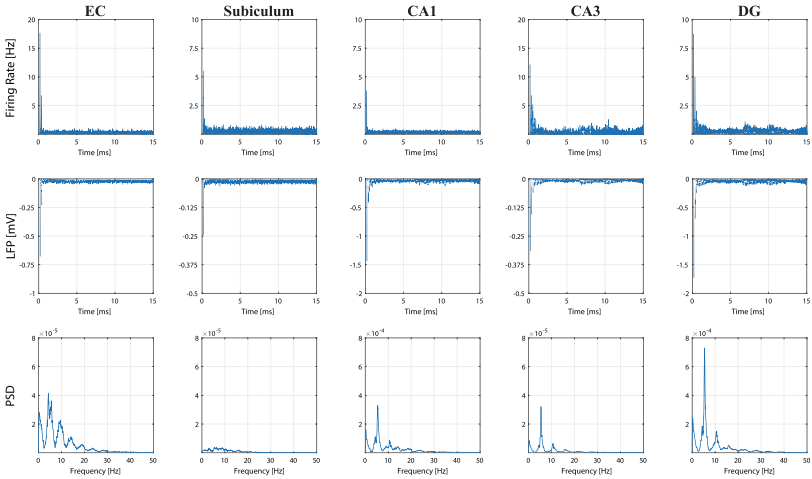


Fig. 2. From top to bottom: the average firing rate of the excitatory neurons population in each sub-region; the local field potential (LFP) measured in the center of each sub-region; the power spectrum density (PSD), obtained from the firing rate of excitatory neurons, for each sub-region.

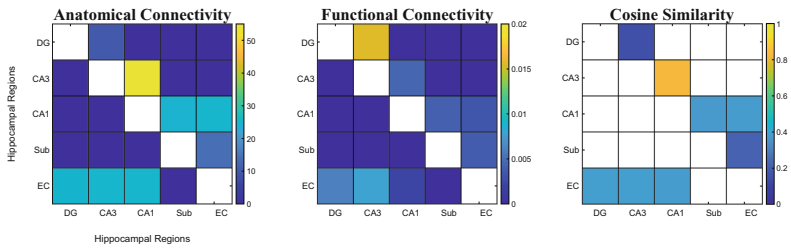


Fig. 3. The anatomical and functional connectivities between the five simulated sub-regions of the hippocampus. The functional connectivity was not measured for areas that do not share direct anatomical connection. To the right, the cosine similarity between the two connectivity matrices. Areas where there was no anatomical connection were excluded from the analysis.

4 Discussion

The novel hippocampal model presented here is grounded in physiological data, and balances a great level of detail, such as including anatomically accurate densities as well as feed-forward and feedback loops characteristic of the hippocampus. Despite this, the model is computationally efficient, enabling longer and repeated simulations necessary for exploring various conditions, such as healthy versus pathological cases. The model can easily be scaled to larger neural populations. At this level of complexity, we only expect minor differences when scaling the model up, due to an increasing number of neurons located

away from the external surface of the simulated regions, compared to neurons close to these borders, which as a result have lower connectivity. Key features and validations of our model include its realistic firing rates in response to physiological inputs, and its ability to generate spontaneous theta oscillations, a hallmark of hippocampal activity. While other models have provided important insights into hippocampal function [13,22,23], not all have undergone similar validation. Furthermore, because other models typically focus on addressing specific issues, they often pay less attention to the major staples of hippocampal circuitry and computational efficiency.

The flexibility and computational efficiency of our model make it a promising tool for investigating hippocampal function across a variety of contexts. By expanding the model to incorporate plasticity mechanisms and long-term memory formation, it will be possible to study a broader range of cognitive functions and their associated dysfunctions, such as those involved in neurodegenerative diseases and memory impairments. This next phase of development will provide deeper insights into both healthy and pathological conditions, reinforcing the model's potential as a comprehensive platform for future research.

References

1. Strange, B.A., Witter, M.P., Lein, E.S., Moser, E.I.: Functional organization of the hippocampal longitudinal axis. Nat. Rev. Neurosci. **15**(10), 655–669 (2014)
2. Eichenbaum, H., Cohen, N.J.: Can we reconcile the declarative memory and spatial navigation views on hippocampal function? Neuron **83**(4), 764–770 (2014)
3. Amaral, D., Lavenex, P.: Hippocampal neuroanatomy (2007)
4. Van Strien, N.M., Cappaert, N.L.M., Witter, M.P.: The anatomy of memory: an interactive overview of the parahippocampal-hippocampal network. Nat. Rev. Neurosci. **10**(4), 272–282 (2009)
5. Moscovitch, M., Cabeza, R., Winocur, G., Nadel, L.: Episodic memory and beyond: the hippocampus and neocortex in transformation. Annu. Rev. Psychol. **67**(1), 105–134 (2016)
6. O'Keefe, J., Dostrovsky, J.: The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. Brain Res. **34**, 171–175 (1971)
7. Yassa, M.A., Stark, C.E.: Pattern separation in the hippocampus. Trends Neurosci. **34**(10), 515–525 (2011)
8. Squire, L.R., Wixted, J.T.: The cognitive neuroscience of human memory since HM. Annu. Rev. Neurosci. **34**(1), 259–288 (2011)
9. Buzsáki, G.: Neural syntax: cell assemblies, synapsembles, and readers. Neuron **68**(3), 362–385 (2010)
10. Koch, C., Segev, I. (eds.): Methods in Neuronal Modeling: From Ions to Networks. MIT Press, Cambridge (1998)
11. Rolls, E.T., Wirth, S.: Spatial representations in the primate hippocampus, and their functions in memory and navigation. Prog. Neurobiol. **171**, 90–113 (2018)
12. Small, S.A., Schobel, S.A., Buxton, R.B., Witter, M.P., Barnes, C.A.: A pathophysiological framework of hippocampal dysfunction in ageing and disease. Nat. Rev. Neurosci. **12**(10), 585–601 (2011)

13. Cutsuridis, V., Graham, B.P., Cobb, S., Vida, I.: Hippocampal Microcircuits: A Computational Modeler's Resource Book. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-99103-0
14. Azizi, A.H., Wiskott, L., Cheng, S.: A computational model for preplay in the hippocampus. Front. Comput. Neurosci. **7**, 161 (2013)
15. Świetlik, D.: Simulations of learning, memory, and forgetting processes with model of CA1 region of the hippocampus. Complexity **2018**(1), 1297150 (2018)
16. Amunts, K., et al.: BigBrain: an ultrahigh-resolution 3D human brain model. Science **340**(6139), 1472–1475 (2013)
17. DeKraker, J., Lau, J.C., Ferko, K.M., Khan, A.R., Köhler, S.: Hippocampal subfields revealed through unfolding and unsupervised clustering of laminar and morphological features in 3D BigBrain. Neuroimage **206**, 116328 (2020)
18. West, M.J., Gundersen, H.J.G.: Unbiased stereological estimation of the number of neurons in the human hippocampus. J. Comp. Neurol. **296**(1), 1–22 (1990)
19. Jinno, S., Kosaka, T.: Stereological estimation of numerical densities of glutamatergic principal neurons in the mouse hippocampus. Hippocampus **20**(7), 829–840 (2010)
20. Stimberg, M., Brette, R., Goodman, D.F.: Brian 2, an intuitive and efficient neural simulator. Elife **8**, e47314 (2019)
21. Pospischil, M., Toledo-Rodriguez, M., Monier, C., Piwkowska, Z., Bal, T., Frégnac, Y., Markram, H., Destexhe, A.: Minimal Hodgkin-Huxley type models for different classes of cortical and thalamic neurons. Biol. Cybern. **99**, 427–441 (2008)
22. Traub, R.D., Miles, R.: Neuronal Networks of the Hippocampus, vol. 777. Cambridge University Press, Cambridge (1991)
23. Aussel, A., Buhry, L., Tyvaert, L., Ranta, R.: A detailed anatomical and mathematical model of the hippocampal formation for the generation of sharp-wave ripples and theta-nested gamma oscillations. J. Comput. Neurosci. **45**(3), 207–221 (2018). https://doi.org/10.1007/s10827-018-0704-x
24. Knowles, W.D., Schwartzkroin, P.A.: Local circuit synaptic interactions in hippocampal brain slices. J. Neurosci. **1**(3), 318–322 (1981)
25. Ropireddy, D., Scorcioni, R., Lasher, B., Buzsáki, G., Ascoli, G.A.: Axonal morphometry of hippocampal pyramidal neurons semi-automatically reconstructed after in vivo labeling in different CA3 locations. Brain Struct. Funct. **216**, 1–15 (2011)
26. Buckmaster, P.S., Wenzel, H.J., Kunkel, D.D., Schwartzkroin, P.A.: Axon arbors and synaptic connections of hippocampal mossy cells in the rat in vivo. J. Comp. Neurol. **366**(2), 270–292 (1996)
27. Empson, R.M., Heinemann, U.: Perforant path connections to area CA1 are predominantly inhibitory in the rat hippocampal-entorhinal cortex combined slice preparation. Hippocampus **5**(2), 104–107 (1995)
28. Amaral, D.G., Ishizuka, N., Claiborne, B.: Chapter neurons, numbers and the hippocampal network. Prog. Brain Res. **83**, 1–11 (1990)
29. Berzhanskaya, J., Urban, N.N., Barrionuevo, G.: Electrophysiological and pharmacological characterization of the direct perforant path input to hippocampal area CA3. J. Neurophysiol. **79**(4), 2111–2118 (1998)
30. Lawrence, J.J., Grinspan, Z.M., McBain, C.J.: Quantal transmission at mossy fibre targets in the CA3 region of the rat hippocampus. J. Physiol. **554**(1), 175–193 (2004)
31. Henze, D.A., Urban, N.N., Barrionuevo, G.: The multifarious hippocampal mossy fiber pathway: a review. Neuroscience **98**(3), 407–427 (2000)

32. Li, X.G., Somogyi, P., Ylinen, A., Buzsáki, G.: The hippocampal CA3 network: an in vivo intracellular labeling study. J. Comp. Neurol. **339**(2), 181–208 (1994)
33. Miles, R.: Synaptic excitation of inhibitory cells by single CA3 hippocampal pyramidal cells of the guinea-pig in vitro. J. Physiol. **428**(1), 61–77 (1990)
34. Xu, X., Sun, Y., Holmes, T.C., López, A.J.: Noncanonical connections between the subiculum and hippocampal CA1. J. Comp. Neurol. **524**(17), 3666–3673 (2016)
35. Rolls, E.T., Deco, G., Huang, C.C., Feng, J.: The effective connectivity of the human hippocampal memory system. Cereb. Cortex **32**(17), 3706–3725 (2022)
36. Zeldenrust, F., Calcini, N., Yan, X., Bijlsma, A., Celikel, T.: The tuning of tuning: How adaptation influences single cell information transfer. PLoS Comput. Biol. **20**(5), e1012043 (2024)
37. Magee, J.C., Cook, E.P.: Somatic EPSP amplitude is independent of synapse location in hippocampal pyramidal neurons. Nat. Neurosci. **3**(9), 895–903 (2000)
38. Bartos, M., Vida, I., Jonas, P.: Synaptic mechanisms of synchronized gamma oscillations in inhibitory interneuron networks. Nat. Rev. Neurosci. **8**(1), 45–56 (2007)
39. MacVicar, B.A., Dudek, F.E.: Local synaptic circuits in rat hippocampus: interactions between pyramidal cells. Brain Res. **184**(1), 220–223 (1980)
40. Bernard, C., Wheal, H.V.: Model of local connectivity patterns in CA3 and CA1 areas of the hippocampus. Hippocampus **4**(5), 497–529 (1994)
41. Miles, R., Traub, R.D., Wong, R.K.: Spread of synchronous firing in longitudinal slices from the CA3 region of the hippocampus. J. Neurophysiol. **60**(4), 1481–1496 (1988)
42. Li, X.G., Somogyi, P., Tepper, J.M., Buzsáki, G.: Axonal and dendritic arborization of an intracellularly labeled chandelier cell in the CA1 region of rat hippocampus. Exp. Brain Res. **90**, 519–525 (1992)
43. Miles, R., Wong, R.K.: Unitary inhibitory synaptic potentials in the guinea-pig hippocampus in vitro. J. Physiol. **356**(1), 97–113 (1984)
44. Thomson, A.M., Radpour, S.: Excitatory connections between CA1 pyramidal cells revealed by spike triggered averaging in slices of rat hippocampus are partially NMDA receptor mediated. Eur. J. Neurosci. **3**(6), 587–601 (1991)
45. Lacaille, J.C., Schwartzkroin, P.A.: Stratum Lacunosum-moleculare interneurons of hippocampal CA1 region. II. Intrasomatic and intradendritic recordings of local circuit synaptic interactions. J. Neurosci. **8**(4), 1411–1424 (1988)
46. Sammons, R.P., Tzilivaki, A., Schmitz, D.: Local microcircuitry of pas shows distinct and common features of excitatory and inhibitory connectivity. Cereb. Cortex **32**(1), 76–92 (2022)
47. Böhm, C., Peng, Y., Geiger, J.R.P., Schmitz, D.: Routes to, from and within the subiculum. Cell Tissue Res. **373**(3), 557–563 (2018). https://doi.org/10.1007/s00441-018-2848-4
48. Winterer, J., et al.: Excitatory microcircuits within superficial layers of the medial entorhinal cortex. Cell Rep. **19**(6), 1110–1116 (2017)
49. Nilssen, E.S., et al.: Inhibitory connectivity dominates the fan cell network in layer II of lateral entorhinal cortex. J. Neurosci. **38**(45), 9712–9727 (2018)
50. Scharfman, H.E., Kunkel, D.D., Schwartzkroin, P.A.: Synaptic connections of dentate granule cells and hilar neurons: results of paired intracellular recordings and intracellular horseradish peroxidase injections. Neuroscience **37**(3), 693–707 (1990)
51. Espinoza, C., Guzman, S.J., Zhang, X., Jonas, P.: Parvalbumin+ interneurons obey unique connectivity rules and establish a powerful lateral-inhibition microcircuit in dentate gyrus. Nat. Commun. **9**(1), 4605 (2018)

52. Csicsvari, J., Hirase, H., Czurkó, A., Mamiya, A., Buzsáki, G.: Oscillatory coupling of hippocampal pyramidal cells and interneurons in the behaving rat. J. Neurosci. **19**(1), 274–287 (1999)
53. Mizuseki, K., Buzsáki, G.: Preconfigured, skewed distribution of firing rates in the hippocampus and entorhinal cortex. Cell Rep. **4**(5), 1010–1021 (2013)

Diffusion Epidemics

Component Structure-Driven Centrality Ranking for Diffusion Management in Complex Networks

Issa Moussa Diop[1](✉), Cherif Diallo[2], and Hocine Cherifi[3]

[1] Université Cote d'Azur, DS4H, I3S, Nice, France
issa-moussa.diop@univ-cotedazur.fr
[2] Université Gaston Berger, LACCA LAB, Saint-Louis, Senegal
cherif.diallo@ugb.edu.sn
[3] Université De Bourgogne, ICB, UMR 6303 CNRS, Dijon, France
hocine.cherifi@u-bourgogne.fr

Abstract. This paper introduces a component-aware ranking strategy to identify critical nodes in modular networks. It overcomes the limitations of traditional centrality-based methods that fail to account for mesoscopic complex network structuration. The component structure decomposes networks into local (densely connected areas) and global (bridges between these areas) components allowing for the flexible use of different centrality measures in each component to better identify key nodes. We evaluate the method using the Susceptible-Infected-Recovered (SIR) model on LFR synthetic networks that simulate varying levels of community strength, ranging from tightly knit to highly mixed communities. The experimental results demonstrate that the community-aware strategy significantly improves diffusion containment in networks with strong community structures compared to classical centrality ranking methods. Its performance is less pronounced in moderately modular networks, and in weakly modular networks, its benefits are minimal. Additionally, we explore the impact of combining centrality measures (e.g., Betweenness for global and Degree for local components), finding marginal improvements in moderately mixed networks but no substantial advantages in highly mixed ones. Our findings suggest that the component-aware strategy is particularly suited for networks with pronounced community structures, offering a robust tool for information-spreading applications where targeting influential nodes is crucial.

Keywords: Component Structure · Diffusion · Centrality · Ranking · SIR model

1 Introduction

Interconnected entities comprise computer, biological, social, and financial systems, allowing a virus, rumor, or information to spread. This phenomenon can

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 53–65, 2025.
https://doi.org/10.1007/978-3-031-82439-5_5

be stopped or accelerated by targeting certain vital entities. To solve this issue, Influence Maximization intends to choose a minimal set of individuals, called a seed set, susceptible to maximizing the spread of influence under an information diffusion process[1–3]. In contrast, immunization targets the seed set to contain the propagation[4,5]. Although the optimal node sets for the two problems often overlap significantly, they are not equivalent[6].

As these problems are NP-Hard, heuristics are sought after because to provide scalability and speed for large-scale or time-sensitive applications. Centrality ranking is a popular heuristic for identifying influential nodes in a network. Relying on local or global topological properties, it offers a quick approximation without modeling the diffusion process. As a result, it often fails to capture the collective influence of multiple nodes, leading to suboptimal spread compared to algorithms specifically designed for influence maximization. The process of centrality ranking involves calculating centrality measures that assess the importance or influence of nodes within a network. Nodes are then ranked according to their centrality scores, with higher-ranked nodes considered more influential. For diffusion, the top nodes are chosen as seeds to maximize spread, while for immunization, the top nodes are selected to minimize spread.

Conventional centrality measures do not exploit the community structure of networks, which can play a crucial role in diffusion dynamics. Several papers have proposed community-based strategy methods that take this aspect into account. Masuda [7] was the first to introduce a centrality measure for the immunization of modular networks. The mod centrality relies on the coarse-gaining network that represents connectivity among communities. This method targets nodes with important inter-community links. They investigate synthetic and real-world networks. Results show the effectiveness of the Mod centrality in modular networks. In [8], the authors propose a community-based centrality. This strategy considers strong links that characterize edges between nodes belonging to the same community and weak links that define edges formed by two nodes belonging to different communities. Considering the edge properties and the communities' sizes determine the importance of nodes. [9] presents a centrality measure based on the community structure. Indeed, a node has a local influence (hubs within a community) and a global influence (bridge nodes that connect different communities). The measure combines these two types of links to target nodes in an immunization process. Experiments consider the SIR epidemic spreading model to test the immunization strategy. Results indicate that this approach is more effective than traditional centrality measures. Building on this work, [10] presents three deterministic immunization strategies: the number of neighboring communities measure, the community hub-bridge measure, and the weighted community hub-bridge measure. The first strategy prioritizes nodes with multiple connections to different communities. The second strategy focuses on nodes that act as hubs within large communities and connect to several other communities. The third strategy adjusts the community hub-bridge measure by weighting each term according to the strength of the community structure. Numerical simulations using the SIR model on real-world and synthetic networks

demonstrate that this method is effective and outperforms stochastic and deterministic strategies designed for modular networks. Nowadays, there is a great deal of work on the so-called community-aware centrality measures. For more information, the reader can refer to [11–24].

Although community-aware centrality measures incorporate the knowledge of the mesoscopic organization of the network to compute the centrality measure, they use the classical ranking strategy to target nodes in decreasing order of the centrality value. Recently, an alternative approach has been proposed. It uses classical centrality measures combined with a community-aware ranking strategy [25]. This work is in line with this approach. We propose a ranking strategy that relies on a mesoscopic decomposition of a network called the component structure. Indeed, one can extract local components and global components from a network. The local components are the dense parts of the network. Nodes and links that participate in the connection between these dense parts form the global components. The component structure is a promising representation that has shown effectiveness in analyzing complex networks with a mesoscopic organization [26–28]. The component structure allows more flexibility than alternative mesoscopic representations such as the community or multi-core periphery structures. Incorporating the mesoscopic representation of networks in the ranking strategy using the component structure is advantageous. Indeed, one can use tailored centrality measures on local and global components before ranking. To evaluate the proposed ranking strategy, we consider the classical scenario of a diffusion process based on the SIR model. Note that the results must be taken with caution because they can fluctuate according to the complexity of the diffusion model under evaluation[29]

The remainder of this paper is structured as follows: Sect. 2 introduces the proposed ranking strategy. Section 3 details the data and methods. Section 4 presents the results of our comparative analysis. Finally, Sect. 5 concludes the paper.

2 Proposed Ranking Strategy

Rajeh et al. propose a community-aware centrality ranking strategy in [25]. They propose to rank nodes in each community independently in decreasing order of a given centrality measure. For a given budget of nodes, the overall ranking strategy consists of ranking the top nodes sequentially according to the decreasing order of the community size. This process is iterated until the budget of nodes is reached. Unlike the classical descending order ranking scheme, which may focus on nodes within a few communities and overlook large parts of the network, this method targets nodes across all regions of the networks. The main drawback of this strategy is that it does not distinguish between inter-community and intra-community links.

The proposed ranking method is based on the component structure. It divides the network into sub-networks called components. One distinguishes two types of components. The local components represent densely connected areas of the

network. The global components bridge these dense areas. If we refer to the community structure, local components are the communities with the intra-community links, and the global components contain the inter-community links and nodes. Therefore one can process the components accordingly. Indeed, centrality measures can be calculated separately for the local and global components or applied uniformly across both types. This flexible approach allows for a more nuanced identification of influential nodes within the network's structure. The algorithm proceeds as follows:

1. Extraction of the component structure (local and global components)
2. Rank the local components from the largest to the smallest size
3. Rank the global components from the largest to the smallest size
4. In each global component, rank nodes using a centrality measure α
5. In each global component, rank nodes using a centrality β
6. Repeat
   (a) Select in each local component the top node
   (b) Select in each global component the top node
7. Until the budget is reached

This strategy is local and global [30]. Note that a node can belong to both local and global components. A node targeted in the global components cannot be targeted in the local components. In addition, one can use the same centrality measure for both types of components.

2.1 Toy Example

Figure 1 illustrates the proposed ranking method on a network with 29 nodes. In this example, the network contains four dense parts (C1 to C4). Thus, there are four local components (LC1 to LC4) when we remove all the links between these dense parts and three global components (GC1 to GC3) when we remove all the links within the dense parts.

Figure 1 A displays the traditional global ranking based on Degree, where the six most important nodes respectively $V_{14}, V_9, V_{15}, V_{25}, V_{26}$, and V_{22}, highlighted in green, are primarily concentrated in specific regions (C1 and C2). This ranking fails to recognize key nodes in other regions like C3 and C4, resulting in an unbalanced distribution of influential nodes across the network.

Figure 1.B illustrates our ranking approach. It separates the network into four local components (LC) and three global components (GC), where key nodes are identified within their respective substructures, based on Degree. The blue nodes represent the 6 top-ranking nodes ($V_9, V_{26}, V_{19}, V_2, V_{14}$, and V_{29}) across both local and global components. Indeed, one first selects the highest hub in each local component. Secondly, one targets the highest hub in each global component. Note that we don't consider GC3 because of its small size. This approach ensures a more balanced representation of influential nodes throughout the network. It highlights important nodes in underrepresented regions (C3 and C4) that would be overlooked by a global-only ranking system.

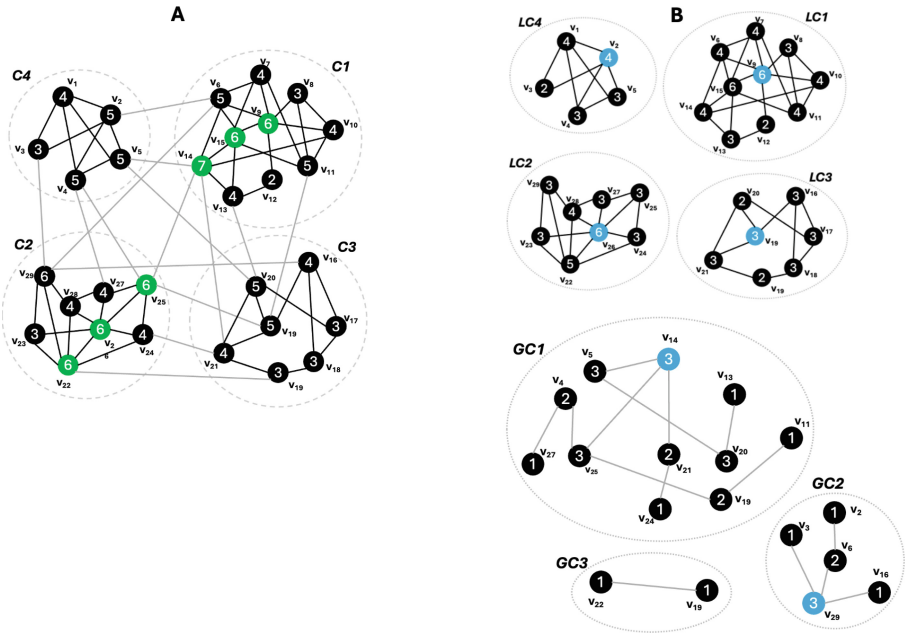


Fig. 1. Comparison of top nodes identified using the descending order ranking based on Degree and our ranking strategy in a network with four local components (LC1 to LC4) and three global components (GC1 to GC3). (A) shows the top six nodes based on a degree ranking strategy, where nodes with the highest degree are colored in green. (B) presents our ranking approach, where the top six nodes are identified separately in the global and local components. The top nodes in the global components are highlighted in blue.

The comparison between the two figures highlights how different ranking methods can emphasize distinct nodes as the most critical, depending on whether a purely global or this strategy perspective is applied. This demonstrates how our approach captures vital nodes not only based on their degree but also considering their roles within both local and global network components.

3 Data and Methods

3.1 Data

To evaluate the proposed approach, we generate synthetic networks using the LFR algorithm [31,32]. It generates synthetic networks with realistic features, including power-law distributed degree and community size. It allows controlling the community structure strength through the mixing parameter (μ). Low μ values result in well-separated communities, while larger values lead to more interconnected communities. Table 1 reports the parameter values used in the experiments. The mixing parameter varies from a strong community to a weak community structure.

Table 1. Caption text

LFR parameter	Values
Number of nodes	2000
Average degree	7
Exponent for the degree distribution	2.7
Exponent for the community size distribution	2.7
Mixing parameter	[0.05, 0.25, 0.65]
Minimum community size	6
Maximum community size	200

3.2 Method

We conduct two sets of experiments. First, we evaluate the effectiveness of the proposed ranking strategy against the classical ranking strategy using the SIR (Susceptible-Infected-Recovered) model and a single centrality measure α. Nodes are ranked in descending order based on these measures, and the top-ranked nodes are removed from the network. To assess the impact of this removal, we simulate the spread of an epidemic using the Susceptible-Infected-Recovered (SIR) model, recording the final outbreak size once the simulation reaches a steady state. In Parallel, we perform the proposed ranking approach with a single centrality measure α. Based on the community structure, we extract the global and local components from the LFR networks. Nodes are ranked according to the proposed ranking strategy, which accounts for global and local importance using the centrality measure α independently in the components.

Figure 2 presents an overview of comparing a standard centrality-based node ranking method α and the proposed method using the same centrality measure. Nodes are ranked based on this centrality, and the top k nodes are infected using a SIR (Susceptible-Infected-Recovered) model. Simultaneously, the proposed method extracts the global and local components from the network. The centrality measure α is computed separately for each component. The proposed ranking method is applied. Infecting the top k nodes, we run the SIR model and calculate the outbreak size. The evaluation metric ΔR is then used to assess the method's performance. The second set of experiments investigates using different centrality measures in the local and global components. Indeed, as the components are networks, we face two situations. Either we calculate the same centrality for each type of component or a different measure of centrality for each type of component. Thus, using two standard centrality measures (α,β) we can create four different types of rankings (α local, α global; β local, β global; α local, β global; β local, α global). Based on these measures, nodes are ranked according to the proposed ranking strategy, which accounts for global and local importance. Figure 3 illustrates the comparison between the proposed method using a single centrality measure and the method which combines two centrality measures. It starts with LFR networks, where both methods extract global and

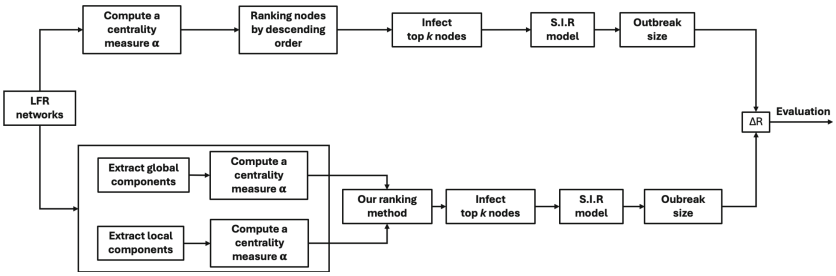


Fig. 2. A schematic representation of the baseline method and the standard centrality for ranking nodes in LFR networks. Both methods compute the centrality measure α. The standard method ranks nodes globally. The proposed method extracts both global and local components, applies α separately to each, and ranks the nodes accordingly. Both methods are evaluated using ΔR, a metric based on the outbreak size calculated from the SIR model.

local components. The single centrality method computes the centrality measure α for all components independently and ranks nodes based on it. The proposed method introduces two centrality measures, α for global components and β for local components. After ranking nodes using this dual-measure approach, the top k nodes are infected, and we run the SIR model to determine the outbreak size. The evaluation metric ΔR is then used for comparison.

To assess the performance of the proposed ranking strategy against an alternative, we compare the outbreak size. The relative difference in outbreak size, ΔR, is defined as $\Delta R = \frac{R_1 - R_0}{R_1}$, where R_1 represents the outbreak size using the proposed strategy, and R_0 represents the outbreak size using a classical centrality method. A positive ΔR value indicates that our method is more effective than the classical methods, while a negative value suggests the opposite.

4 Results

4.1 Comparing to Classical Ranking Strategy

Figure 4 displays the relative outbreak size as a function of the fraction of initially infected nodes with strong, medium, and weak community structure strengths. The curves indicate the relative performance difference of the proposed ranking strategy with the descending order ranking for the Degree centrality measure. In this ranking method, we compute the Degree in each component. At a low mixing parameter, our ranking method scheme significantly outperforms the classical scheme, as indicated by the consistently positive relative outbreak size values. Indeed, The difference relative outbreak size curve reaches its peak (around 8%) when approximately 8% of the nodes are initially infected. After, the curve decreases until it reaches 4%. In a network with a medium community structure strength, the advantage of our ranking scheme decreases. Indeed, ΔR is very low when the fraction of infected nodes is less than 10%. After this value, as

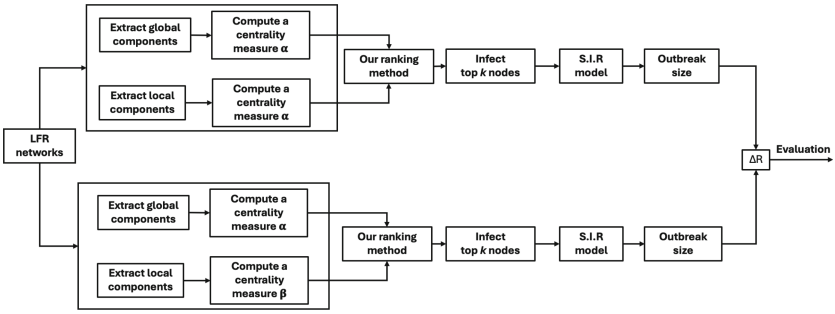


Fig. 3. A schematic representation of the baseline method and the method based on combining two centrality measures for ranking nodes in LFR networks. The baseline method applies the centrality measure α to all components, whereas the proposed method introduces two different centrality measures: α for global components and β for local components. After ranking nodes based on these measures, the top k nodes are infected, and the SIR model is run to determine the outbreak size. Both methods are evaluated using the ΔR metric, reflecting the difference in outbreak sizes.

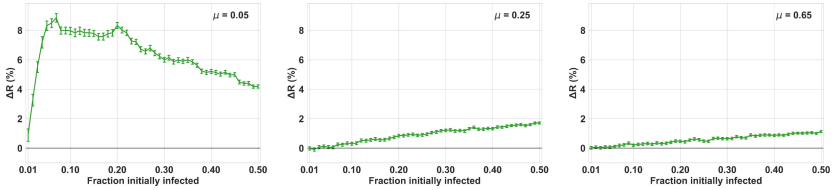


Fig. 4. Variation of outbreak size ΔR between the ranking based on the baseline method and the ranking based on Degree centrality as a function of the fraction of initially infected nodes for different values of infection probability μ= 0.05, 0.25, 0.65. The baseline method applies Degree centrality in both the local and global components

the fraction of infected nodes increases, the increase in ΔR is more gradual and steady compared to the situation of the low mixing parameter. The maximum of the curve is near 2% and is reached when 50% are infected. At a high mixing parameter, the change in ΔR is smaller across the range of initially infected fractions. This trend is similar when the network has a medium community structure.

Figure 5 shows the relative outbreak size as a function of the fraction of initially infected nodes with strong, medium, and weak community structure strengths. The curves indicate the relative performance difference of the component-aware ranking strategy with the descending order ranking for the Betweenness centrality measure. In this component-aware ranking, we compute the Betweenness in each component.

In the network with a strong community structure strength, the component-ranking strategy demonstrates a strong initial performance, with ΔR values

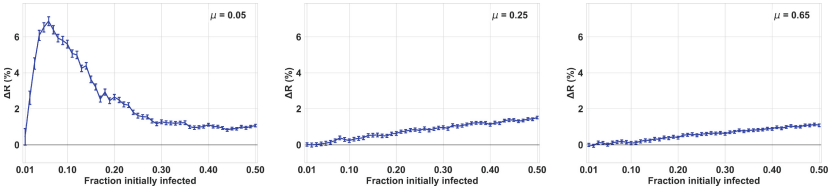


Fig. 5. Variation of outbreak size ΔR between the ranking based on the baseline method and the ranking based on Betweenness centrality as a function of the fraction of initially infected nodes for different values of infection probability μ= 0.05, 0.25, 0.65. The baseline method applies Betweenness centrality in both the local and global components

peaking around 7% when a small fraction (less than 10% of the node) of the network is initially infected. However, as the fraction of initially infected nodes increases, the relative outbreak size diminishes rapidly, and trends towards 1%. With a moderate mixing parameter, the effectiveness of our ranking strategy remains constant for all the fractions of infected nodes. As the rate of infection increases, so does the ΔR and reach around 2%. We observe the same behaviors for high mixing parameter value. One can see that the relative outbreak size values are close to zero, with a slight positive trend as the fraction of initially infected nodes increases.

4.2 Comparing the Component-Aware Ranking Strategy

The component-aware ranking offers 4 types of ranking strategies using two centrality measures (α,β). The first two ranking use the same centrality measure in the local and the global components (α or β). We call it the baseline strategy. The second ranking uses different centrality measures in the local and global components. This subsection investigates these strategies.

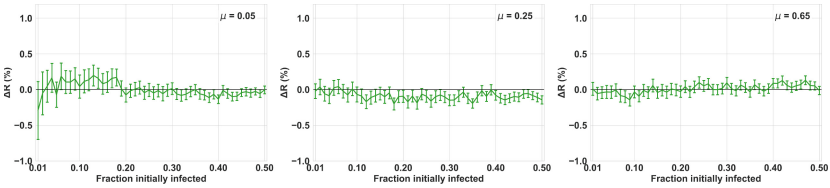


Fig. 6. Comparison of the performance of two ranking strategies for different values of the mixing parameter. The first strategy (baseline) applies Degree centrality in both the local and global components, while the second strategy combines Betweenness centrality in the global component with Degree centrality in the local components. ΔR represents the difference in outbreak sizes between the two strategies.

Figure 6 compares two component-aware ranking strategies based on the difference in outbreak size across varying fractions of initially infected nodes. The first strategy, referred to as the baseline strategy, uses Degree centrality in both the global and local components of the network. The second strategy applies Betweenness centrality in the global component and Degree centrality in the local components. When the mixing parameter is low, the relative difference outbreak size values show more significant fluctuations, with some points rising slightly above or below 0%. The curve shows that the baseline strategy is more effective when the fraction of infected nodes is smaller than the combined ranking strategy. In the network with a medium community structure, ΔR remains near zero, but with slightly lower variability compared to $\mu = 0.05$. However, the combined strategy is more efficient over the fraction of infected nodes. In networks with high mixing, ΔR remains close to zero with very few deviations. This indicates that at higher mixing levels, the combined ranking method offers no significant advantage over the baseline.

Overall, for all values of the mixing parameters, the combined ranking scheme provides similar results to the baseline (based on Degree), as indicated by the relative outbreak size hovering around zero.

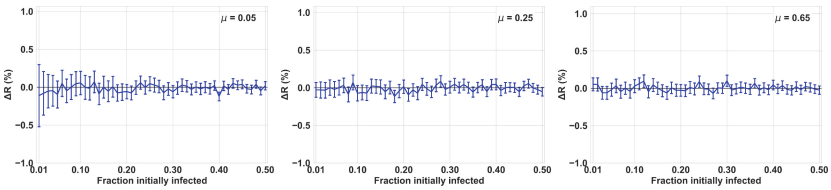


Fig. 7. Comparison of the performance of two ranking strategies for different values of the mixing parameter. The first strategy (baseline) applies Betweenness centrality in both the local and global components, while the second strategy combines Betweenness centrality in the global component with Degree centrality in the local components. ΔR represents the difference in outbreak sizes between the two strategies.

Figure 7 compares two component-aware ranking strategies based on the difference in outbreak size across varying fractions of initially infected nodes. The first strategy, referred to as the baseline strategy, uses Betweenness centrality in both the global and local components of the network. The second strategy remains the same.

We observe that both strategies (applying Betweenness in both the global and local components, and the combined strategy where Betweenness is used in the global component and Degree in local components) yield very similar results. The ΔR values fluctuate around 0%, indicating that there is little to no significant difference in the performance of the two strategies in terms of outbreak size. Indeed, For $\mu = 0.05$, where the network is more clustered and less mixed, the fluctuations in ΔR are more pronounced, though the overall impact remains close to zero. As the mixing parameter increases ($\mu = 0.25$ and

$\mu = 0.65$), these fluctuations become smaller, suggesting that the influence of either ranking strategy diminishes in more homogeneous or well-mixed networks.

5 Conclusion

This paper presents a component-aware ranking strategy for identifying influential nodes in modular networks. This method addresses the limitations of traditional global ranking methods, which doesn't often consider the mesoscopic structure of networks. To validate the effectiveness of the proposed approach, we conducted experiments using the SIR model to simulate epidemic spreading on LFR synthetic networks. These networks with varying levels of mixing parameters, allowing us to assess the performance of the proposed method in both strongly and weakly modular networks. We consider a baseline method in which we compute a centrality measure in the local and the global components. This baseline method is compared against classical centrality measures, such as Degree and Betweenness centrality measures, to assess its ability to control epidemic outbreaks. In addition, the baseline method is evaluated by applying different combinations of centrality measures to the global and local components extracted from the network.

The results show that the component-aware ranking strategy consistently outperformed classical ranking methods in networks with strong community structures strength. In networks with medium community structure strength, the advantage of the component-aware ranking remains positive but less pronounced. At high mixing parameters, indicative of weak community structures, the performance gains of the community-aware method remains minimal. Therefore, for diffusion processes, applying the same centrality measure (either Degree or Betweenness) to both local and global components is more effective than using Degree or Betweenness across the entire network.

Furthermore, we compare two component-aware ranking strategies using the same centrality measure for both local and global components (baseline) versus combining different centrality measures (e.g., Betweenness for global and Degree for local components). We find that the combined strategy offered marginal improvements in networks with medium mixing parameters. Specifically, the combined approach is slightly more effective as the fraction of initially infected nodes increases, particularly against baseline in which the Degree is used in the local and global components. However, in highly mixed networks, the combined strategy showed no significant advantage over the baseline.

In future work, we plan to expand our research by exploring the effects of additional LFR parameters. We will also apply our approach to real-world networks and investigate other centrality measures. Furthermore, we aim to enhance the ranking algorithm by optimizing parameters such as the distribution of the node budget between local and global components, as well as refining the combination of centrality measures used in the ranking process.

References

1. Jaouadi, M., Romdhane, L.B. : A survey on influence maximization models. Expert Syst. Appl. **248**, 123429 (2024)
2. Banerjee, S., Jenamani, M., Pratihar, D.K.: A survey on influence maximization in a social network. Knowl. Inf. Syst. **62**(9), 3417–3455 (2020). https://doi.org/10.1007/s10115-020-01461-4
3. Morone, F., Makse, H.A.: Influence maximization in complex networks through optimal percolation. Nature **524**(7563), 65–68 (2015). https://doi.org/10.1038/nature14604
4. Zino, L., Cao, M.: Analysis, prediction, and control of epidemics: a survey from scalar to dynamic network models. IEEE Circ. Syst. Mag. **21**(4), 4–23 (2021)
5. Rosenblatt, S.F., Smith, J.A., Gauthier, G.R., Hébert-Dufresne, L.: Immunization strategies in networks with missing data. PLoS Comput. Biol. **16**(7), e1007897 (2020). https://doi.org/10.1371/journal.pcbi.1007897
6. Radicchi, F., Castellano, C.: Fundamental difference between superblockers and superspreaders in networks. Phys. Rev. E **95**(1), 012318 (2017)
7. Masuda, N.: Immunization of networks with community structure. New J. Phys. **11**(12), 123018 (2009)
8. Zhao, Z., Wang, X., Zhang, W., Zhu, Z.: A community-based approach to identifying influential spreaders. Entropy **17**(4), 2228–2252 (2015)
9. Gupta, N., Singh, A., Cherifi, H.: Centrality measures for networks with community structure. Phys. A: Stat. Mech. Appl. **452**, 46–59 (2016)
10. Ghalmane, Z., Hassouni, M.E., Cherifi, H.: Immunization of networks with non-overlapping community structure. Soc. Netw. Anal. Min. **9**(1), 1–22 (2019). https://doi.org/10.1007/s13278-019-0591-9
11. Magelinski, T., Bartulovic, M., M. Carley, K.: Measuring node contribution to community structure with modularity vitality. IEEE Trans. Netw. Sci. Eng. **8**(1), 707–723 (2021). https://doi.org/10.1109/TNSE.2020.3049068
12. Blöcker, C., Nieves, J.C., Rosvall, M.: Map equation centrality: community-aware centrality based on the map equation. Ap. Net. Sci. **7**(1), 56 (2022)
13. Gupta, N., Singh, A., Cherifi, H.: Community-based immunization strategies for epidemic control. In: 2015 7th International Conference on Communication Systems and Networks (COMSNETS), pp. 1–6. IEEE (2015)
14. Lv, L., et al.: A community-based centrality measure for identifying key nodes in multilayer networks. IEEE Trans. Comput. Social Syst. **11**(2), 24448–2463 (2023)
15. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: Exploring hubs and overlapping nodes interactions in modular complex networks. IEEE Access **8**, 79650–79683 (2020)
16. Saxena, R., Kaur, S., Bhatnagar, V.: Social centrality using network hierarchy and community structure. Data Min. Knowl. Disc. **32**(5), 1421–1443 (2018). https://doi.org/10.1007/s10618-018-0582-x
17. Rajeh, S., Savonnet, M., Leclercq, E., Cherifi, H.: Interplay between hierarchy and centrality in complex networks. IEEE Access **8**, 129717–129742 (2020)
18. Rajeh, S., Savonnet, M., Leclercq, E., Cherifi, H.: Characterizing the interactions between classical and community-aware centrality measures in complex networks. Sci. Rep. **11**(1), 10088 (2021)
19. Rajeh, S., Savonnet, M., Leclercq, E., Cherifi, H.: Comparing community-aware centrality measures in online social networks. In: Mohaisen, D., Jin, R. (eds.) Computational Data and Social Networks: 10th International Conference, CSoNet 2021,

Virtual Event, November 15–17, 2021, Proceedings, pp. 279–290. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-91434-9_25
20. Rajeh, S., Yassin, A., Jaber, A., Cherifi, H.: Analyzing community-aware centrality measures using the linear threshold model. In: Benito, R.M., Cherifi, C., Cherifi, H., Moro, E., Rocha, L.M., Sales-Pardo, M. (eds.) Complex Networks & Their Applications X: Volume 1, Proceedings of the Tenth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2021, pp. 342–353. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-030-93409-5_29
21. Rajeh, S., Savonnet, M., Leclercq, E., Cherifi, H.. Identifying influential nodes using overlapping modularity vitality. In: Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 257–264 (2021)
22. Rajeh, S., Savonnet, M., Leclercq, E., Cherifi, H.: Comparative evaluation of community-aware centrality measures. Qual. Quan. **57**(2), 1273–1302 (2023)
23. Ghalmane, Z., El Hassouni, M., Cherifi, C., Cherifi, H.: Centrality in modular networks. EPJ Data Sci. **8**(1) (2019). https://doi.org/10.1140/epjds/s13688-019-0195-7
24. Rajeh, S., Cherifi, H.: A community-aware centrality framework based on overlapping modularity. Soc. Netw. Anal. Min. **13**(1), 37 (2023)
25. Rajeh, S., Cherifi, H.: Ranking influential nodes in complex networks with community structure. PLoS ONE **17**(8), e0273610 (2022)
26. Diop, I.M., Cherifi, C., Diallo, C., Cherifi, H.: Revealing the component structure of the world air transportation network. Appl. Netw. Sci. **6**(1), 1–50 (2021). https://doi.org/10.1007/s41109-021-00430-2
27. Diop, I.M., Cherifi, C., Diallo, C., Cherifi, H.: Robustness of the weighted world air transportation network components. In: 2022 IEEE Workshop on Complexity in Engineering (COMPENG), pp. 1–6. IEEE (2022)
28. Hmaida, S., Cherifi, H., El Hassouni, M.: A multilevel backbone extraction framework. Appl. Netw. Sci. **9**(1), 41 (2024)
29. Rajeh, S., Cherifi, H.: On the role of diffusion dynamics on community-aware centrality measures. PLoS ONE **19**(7), e0306561 (2024)
30. Cherifi, H., Palla, G., Szymanski, B.K., Lu, X.: On community structure in complex networks: challenges and opportunities. Appl. Netw. Sci. **4**(1), 1–35 (2019). https://doi.org/10.1007/s41109-019-0238-9
31. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E-Stat., Nonlinear, Soft Matter Phys. **78**(4), 046110 (2008)
32. Orman, G.K., Labatut, V., Cherifi, H.: Towards realistic artificial benchmark for community detection algorithms evaluation. Int. J. Web Based Commun. **9**(3), 349 (2013). https://doi.org/10.1504/IJWBC.2013.054908

The Impact of Heterogeneity on Epidemics: Insights from a Modified SIR Model

Fabio Mazza[1](✉), Francesca Colaiori[2,3], Stefano Guarino[4], Sandro Meloni[4], Marco Brambilla[1], Carlo Piccardi[1], Francesco Pierri[1], and Fabio Saracco[4,5,6]

[1] Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan, Italy
fabio.mazza@polimi.it
[2] Istituto dei Sistemi Complessi, Consiglio Nazionale delle Ricerche, Roma, Italy
[3] Dipartimento di Fisica, Università di Roma La Sapienza, Roma, Italy
[4] Istituto per le Applicazioni del Calcolo "Mauro Picone", Consiglio Nazionale delle Ricerche, Roma, Italy
[5] Centro Studi e Ricerche "Enrico Fermi", Roma, Italy
[6] IMT School For Advanced Studies Lucca, Lucca, Italy

Abstract. Human behavior is a key determinant of epidemic outcomes. During health crises, variations in people's responses to control measures, often driven by different levels of risk perception, lead to variability in epidemic parameters such as infectiousness and susceptibility. We introduce a model within the Susceptible-Infected-Removed (SIR) class that accounts for these heterogeneities. We find that there is a region in the space of parameters just above the epidemic threshold, where trajectories showing an initial decline in the number of Infected can suddenly reverse and give rise to widespread transmission. Such heterogeneity can lead to an underestimation of transmission potential and delayed recognition of epidemic resurgence, thereby severely compromising efforts for a timely response. We examine this phenomenon in the mean-field scenario and then simulate the dynamics on homogeneous and heterogeneous contact networks, confirming that this phenomenology persists beyond mean field. Our model also encompasses cases where the heterogeneity originates from biological or other factors.

Keywords: complex networks · human behavior · epidemics · SIR

1 Introduction

The effect of population heterogeneity in the evolution of an epidemic is widely studied in the scientific literature. Previous work considers heterogeneity in connectivity [14], behavior [5,13], susceptibility [12] or infectivity [3,8,10,11]. The combined effect of heterogeneous susceptibility and infectivity has been studied in [7] by Miller, focusing on the outbreak size and probability for epidemic evolving on a network. Among other findings, Miller estimates the epidemic threshold

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 66–75, 2025.
https://doi.org/10.1007/978-3-031-82439-5_6

imposing that the average transmission rate equals the reciprocal of the average excess degree of the network. In this paper, we show that, under specific conditions, the actual epidemic threshold can differ from the estimate given in [7]. In more detail, we introduce an epidemic model in the Susceptible-Infected-Removed (SIR) class, denoted HeSIR (*Heterogeneous* SIR), that incorporates heterogeneity by assuming a bimodal distribution of susceptibility and infectivity in the population. In the following we attribute heterogeneity to individual behavior.

Such a variability may more generally arise from a range of factors. For instance, it could be due to some biological characteristics of the hosts, or to the lack of compliance with the safety guidelines during a pandemic [4]. The effect of the latter was clearly observed during the unprecedented global spread of SARS-CoV2 [2,9]. Therefore, in the following we label individuals according to their behavior: Ordinary (O) for those who comply with the guidelines, and Misbehaving (M) for those who do not. Given that misbehaving individuals are at a higher risk of transmitting and contracting the disease, we assign to them values of infectiousness and susceptibility increased by a factor $a > 1$ and $b > 1$, respectively. Our approach is similar to the modeling framework introduced in [4], where the lack of compliance with preventive measures is attributed to misinformation.

We estimate the epidemic threshold in a random regular graph (RRG), finding the conditions for the existence of a giant percolation cluster when the link occupation probability corresponds to the total transmission probability for that link.

Direct integration of the mean-field equations and simulations on an RRG show the existence of a region corresponding to a "resurgence zone", where an initial fall in the size of infected population is followed by a sudden growth that leads to an outbreak. In the standard SIR model, the condition $\dot{I}(0) > 0$, where I is the fraction of the infected population, is necessary and sufficient for an outbreak to occur. We show that, in our model, the necessary and sufficient condition corresponds to $\dot{I}_{\text{eff}}(0) > 0$, where $I_{\text{eff}} = aI_M + I_O$ is the effective fraction of the infected population. Standard behavior is recovered as soon as $a = 1$ or $b = 1$ (or both), meaning that this aspect of the epidemic can only be captured by models in which susceptibility and infectivity *both* vary.

Finally, to further support our findings, we consider the evolution of our model on two simple networks characterized, respectively, by homophily and degree heterogeneity. The term homophily is generally used to refer to the tendency of "similar" nodes to be connected. In this case, by similar we intend having the same risk attitude, and, hence, same susceptibility and infectivity. Simulations show that homophilic populations (where O (M) individuals are more likely to be connected with other O (M) individuals) correspond to a wider resurgence zone. A similar phenomenon occurs in heterogeneous networks. In that case, we find that the resurgence zone persists in the limit $a = b = 1$, which corresponds to the standard SIR model, showing that topological heterogeneity plays a role similar to heterogeneity in the epidemic parameters. Our

results indicate that heterogeneity in epidemiological parameters can result in an underestimate of transmission potential, which, initially obscured, could trigger a resurgence later. This delay in recognizing the outbreak can, in turn, dramatically compromise an effective and timely epidemic response.

2 Model Description and Analysis

We fix the timescale by setting the susceptibility of ordinary individuals to $\sigma = 1$. The mean–field (MF) equations for the HeSIR model are the following:

$$\begin{cases} \dot{I}_M &= b\beta S_M(aI_M + I_O) - \gamma I_M = b\beta S_M I_{\text{eff}} - \gamma I_M \\ \dot{I}_O &= \beta S_O(aI_M + I_O) - \gamma I_O = \beta S_O I_{\text{eff}} - \gamma I_O \\ \dot{S}_M &= -b\beta S_M(aI_M + I_O) = -b\beta S_M I_{\text{eff}} \\ \dot{S}_O &= -\beta S_O(aI_M + I_O) = -\beta S_O I_{\text{eff}} \\ \dot{R} &= \gamma(I_M + I_O) \end{cases} \tag{1}$$

where I_M (resp. I_O) is the fraction of misbehaving (ordinarily behaving) infected individuals, S_M (S_O) is the fraction of misbehaving (ordinary) susceptible individuals, and R is the fraction of recovered. The total population is conserved, so that $I_M + I_O + S_M + S_O + R = 1$.

To investigate the behaviour of the HeSIR system in the discrete individual-based case, we start from the case where the contacts between the N individuals in the system can be described by a random regular graph (RRG). We resort to the classic equivalence with a percolation problem and find the conditions that lead to the emergence of a giant percolation cluster. Let p be the fraction of misbehaving population and k the number of edges of each node in the graph, and let x_O and x_M be respectively the average probability that an ordinary vertex and a misbehaving vertex are not connected to the giant percolation cluster via a specific one of their edges. Further, for $i, j \in \{O, M\}$, let ϕ_{ij} be the occupation probability for an edge connecting a susceptible vertex of type i with an infected vertex of type j. x_O and x_M satisfy the system

$$\begin{cases} x_O &= (1-p)(1 - \phi_{OO} + \phi_{OO} x_O^{k-1}) + p(1 - \phi_{OM} + \phi_{OM} x_M^{k-1}) \\ x_M &= (1-p)(1 - \phi_{MO} + \phi_{MO} x_O^{k-1}) + p(1 - \phi_{MM} + \phi_{MM} x_M^{k-1}) \end{cases} \tag{2}$$

In fact: the considered edge leads to an ordinary vertex with probability $1-p$, to a misbehaving one with probability p; the node at the other end of the considered edge has other $k-1$ links.

The system admits a trivial solution for $x_O = x_M = 1$, corresponding to the absence of any giant cluster. The epidemic threshold can thus be found imposing that the two equations in (2) define curves being tangent in $(1, 1)$. To do so, we define

$$f(x_O, x_M) = x_O - (1-p)(1 - \phi_{OO} + \phi_{OO} x_O^{k-1}) - p(1 - \phi_{OM} + \phi_{OM} x_M^{k-1}) \tag{3}$$

$$g(x_O, x_M) = x_M - (1-p)(1 - \phi_{MO} + \phi_{MO} x_O^{k-1}) - p(1 - \phi_{MM} + \phi_{MM} x_M^{k-1}) \tag{4}$$

and we impose $\nabla f(1,1) = \nabla g(1,1)$. Since the occupation probability ϕ_{ij} is the total transmission probability for that edge, we use the approximation $\phi_{OO} \approx \beta/\gamma$, $\phi_{OM} \approx a\beta/\gamma$, $\phi_{MO} \approx b\beta/\gamma$ and $\phi_{MM} \approx ab\beta/\gamma$. Finally, we find

$$\left(\frac{\beta}{\gamma}\right)_c = \frac{1}{(k-1)(abp+1-p)} \tag{5}$$

If, instead, we let x be the average probability that *any* random vertex is connected to the giant percolation cluster via a specific one of its edges, the system (2) is replaced by the standard equation $x = 1 - \phi + \phi x^{k-1}$, where the average occupation probability is

$$\phi = (1-p)^2\phi_{OO} + p(1-p)\phi_{OM} + p(1-p)\phi_{MO} + p^2\phi_{MM} \tag{6}$$

In this case, the condition for the existence of a giant percolation cluster is estimated as

$$\left(\frac{\beta}{\gamma}\right)_* = \frac{1}{(k-1)(ap+1-p)(bp+1-p)} \tag{7}$$

which, for k-regular graphs, coincides with the estimate found in [7].

While (5) is the actual epidemic threshold, the epidemic behavior is described by both critical thresholds $(\beta/\gamma)_c$ and $(\beta/\gamma)_*$: for $\beta/\gamma < (\beta/\gamma)_c$ outbreaks are suppressed; for $\beta/\gamma > (\beta/\gamma)_*$ the contagion grows exponentially right from the start; in the intermediate region where $(\beta/\gamma)_c < \beta/\gamma < (\beta/\gamma)_*$ there exists a deceptive phase in which an initial decay is then reverted and followed by an outbreak. This can be proved assuming $R(0) = 0$ and that a small uniformly distributed fraction ϵ of the population is initially infected, so that we have $I_M(0) = p\epsilon$, $I_O(0) = (1-p)\epsilon$, $S_M(0) = p(1-\epsilon)$, $S_O(0) = (1-p)(1-\epsilon)$. In this case, we have

$$\dot{I}(0) = 0 \iff \frac{\beta}{\gamma} = \frac{1}{(ap+1-p)(bp+1-p)(1-\epsilon)} \tag{8}$$

$$\dot{I}_{\text{eff}}(0) = 0 \iff \frac{\beta}{\gamma} = \frac{1}{(abp+1-p)(1-\epsilon)} \tag{9}$$

If ϵ is small, (8) is essentially equivalent to (7), whereas (9) is essentially equivalent to (5). Condition (8) is generally used to separate the two epidemic regimes, the inactive and the active phase. In the HeSIR model, however, an initial decrease in I_O can be reversed once the more active M population becomes involved in the outbreak.

For this reason, (9) determines, up to a factor $1/(1-\epsilon)$, the actual epidemic threshold $(\beta/\gamma)_c$, as confirmed by numerical integration results (Fig. 1). When $p = 0$ (no misbehaving) or $a = b = 1$ (no increase in misbehaving infectivity and susceptibility) the usual SIR dynamics and epidemic threshold are recovered, with $(\beta/\gamma)_c = 1$. When $p > 0$ and $a = 1$ or $b = 1$ (but not both), (8) and (9) are equivalent and $(\beta/\gamma)_c < 1$. Only when $p > 0$ and both $a > 1$ and $b > 1$, (9) sets a weaker and more accurate condition than (8).

Both the numeric solution of system (1) and extensive simulations on a RRG confirm the above described scenario and the existence of three regimes. For the simulations we generated one RRG with $N = 40000$ nodes and degree $k = 800$, and we used an optimized variant of the Gillespie algorithm [6]. The value of k was chosen to be sufficiently large in order to verify the results of the mean field model, and to avoid a saturation effect which could lead to more complex behavior. To empirically distinguish these regimes, we can focus on the number of stationary points of $I(t)$, which are 0 when $\beta/\gamma < (\beta/\gamma)_c$, 2 when $(\beta/\gamma)_c < \beta/\gamma < (\beta/\gamma)_*$, and 1 when $\beta/\gamma > (\beta/\gamma)_*$. Figures 2 and 3 outline the regions for the three regimes, separated by (8) and (9), in the mean field model and the RRG, respectively.

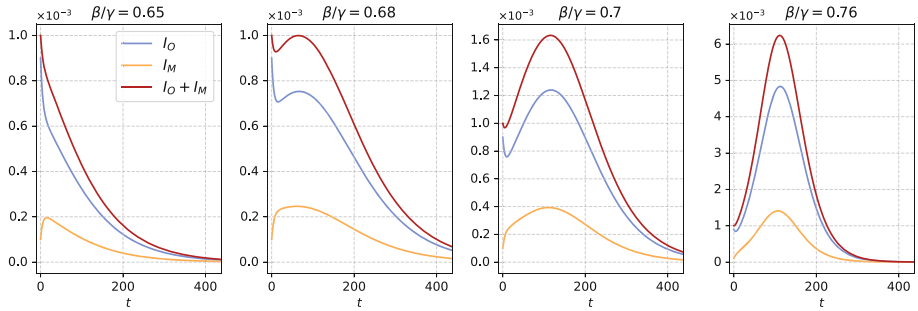


Fig. 1. Fraction of infected individuals in the system, from the integration of the mean field Eqs. (1), for different values of β/γ. The other parameters are set to $\gamma = 0.2$, $a = 2$, $b = 3$ and $p = 0.1$. When β/γ does not satisfy (9), I decreases monotonically to 0 (leftmost plot). On the other hand, when β/γ satisfies (8), I increases right from the start (rightmost plot). Whenever β/γ satisfies (9) but not (8), I initially decreases to then increase and lead to an outbreak (second and third plot).

3 Model Evolution in Heterogeneous Networks

We now investigate whether the phenomenology observed in mean field and on RRG persists on heterogeneous networks. First, we consider the case in which the network is assortative with respect to the vertex type, meaning that the individuals have a preference for connecting with other people with similar behavior (ordinary with ordinary, misbehaving with misbehaving). The level of *homophily* can be controlled by a parameter $h \in [0, 1]$, with the mixing between the two groups defined by the following matrix:

$$(1-h)\begin{pmatrix} 1-p & p \\ 1-p & p \end{pmatrix} + h\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{10}$$

In words, an ordinary (O) vertex connects with another O with probability $(1-p)(1-h)+h$, and with a misbehaving (M) with probability $p(1-h)$; and an

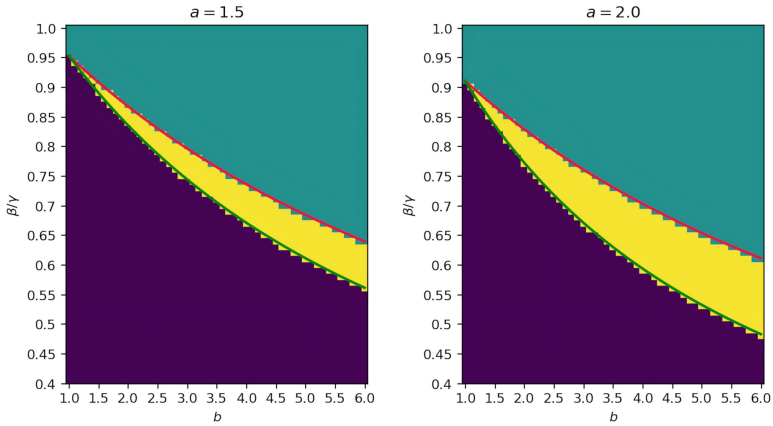


Fig. 2. The number of stationary points (s.p.) in $I(t)$ from numerical integration of the mean field model with different values of β/γ and b (purple: 0 s.p., cyan: 1 s.p., yellow: 2 s.p.). The red curve corresponds to Eq. (7), the green one to Eq. (5). In the yellow region, there is a resurgence of the global fraction of infected population. The initial infected population is $\epsilon = 10^{-3}$, divided among O and M proportionally to the fraction of misbehaving $p = 0.1$, and $\gamma = 0.25$. (Color figure online)

M vertex connects with an O with probability $(1-p)(1-h)$, and with another M with probability $p(1-h)+h$. When $h = 0$ a standard network with no homophily is obtained, while $h = 1$ means that ordinary and misbehaving nodes form two separate connected components, each with a Poisson-like degree distribution as the whole network.

We perform numerical simulations of the HeSIR model on Stochastic Block Model (SBM) graphs of $N = 40000$ individuals, constructed with the mixing matrix above (Eq. 10) with average degree $k = 400$. From the results shown in Figs. 4 and 5, it is clear that when increasing network homophily the region of epidemic resurgence (when 2 stationary points are found in $I(t)$) becomes larger, covering a broader range of β/γ values. In the same figures, we also show the empirical estimate for the epidemic threshold obtained maximizing the epidemic variability V_{AR}, defined as

$$V_{\mathrm{AR}} = \frac{\langle R_\infty^2 \rangle - \langle R_\infty \rangle^2}{\langle R_\infty \rangle}$$

where R_∞ is the attack rate of the epidemic simulations. In both Figs. 4 and 5, this empirical threshold corresponds with the lower boundary of the resurgence zone.

Finally, in order to investigate the effect of the heterogeneity of the degree distribution, we also explore the HeSIR model on Barabàsi-Albert (BA) random graphs. The simulations have been run on a single graph of $N = 40000$ and with each node adding $k = 4$ links with preferential attachment [1]. Figure 6 shows that the resurgence zone is much larger then in the previous cases. As done for

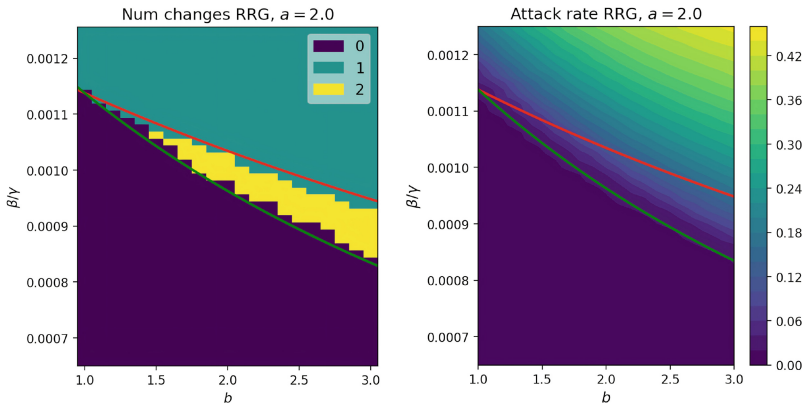


Fig. 3. Results for simulations on RRG graph. Left: number of stationary points of $I(t)$. In the yellow intermediate region $I(t)$ starts decreasing at $t = 0$, then later it increases. Right: Median attack rate (R_∞). The red line corresponds to Eq. (7), the green one to Eq. (5), rescaled by a factor $d-1$. Other parameters are $p = 0.1$, $\gamma = 0.2$.

the SBM, we also show the empirical estimate obtained maximizing the epidemic variability V_{AR}. Again, the empirical epidemic threshold corresponds with the lower boundary of the resurgence zone.

The phenomenon of epidemic resurgence described above can also be found on BA graphs under the SIR model. Its presence in this case can be explained by the heterogeneity of the degree: since most nodes have very low degree, it

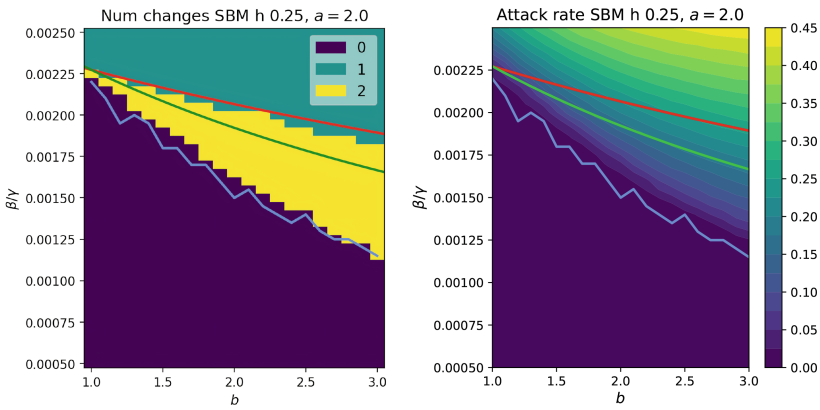


Fig. 4. Results for SBM with $h = 0.25$. Left: number of stationary points of $I(t)$. Right: Median attack rate . The red line corresponds to Eq. (7), the green one to Eq. (5), both rescaled by the average degree of the graph. The light blue lines show the β/γ values corresponding to the peak variability of the attack rate V_{AR}. Other parameters are $p = 0.1$, $\gamma = 0.2$.

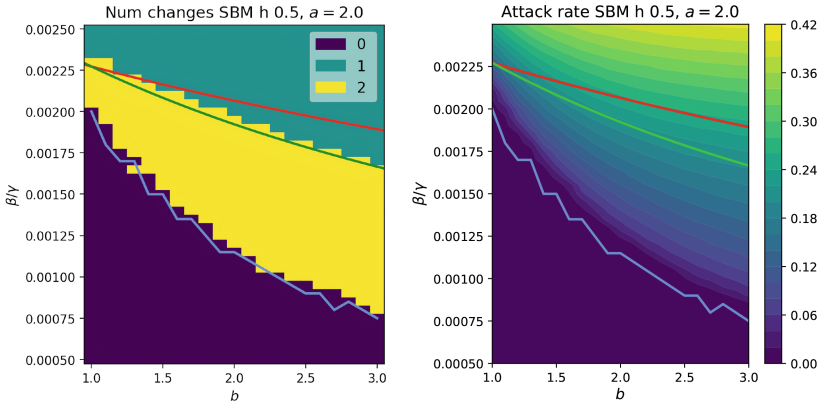


Fig. 5. Results for SBM with $h = 0.5$. Left: number of stationary points of $I(t)$. Right: Median attack rate (R_∞). The red line corresponds to Eq. (7), the green one to Eq. (5), both rescaled by the average degree of the graph. The light blue lines show the β/γ values corresponding to the peak variability of the attack rate V_{AR}. Other parameters are $p = 0.1$, $\gamma = 0.2$.

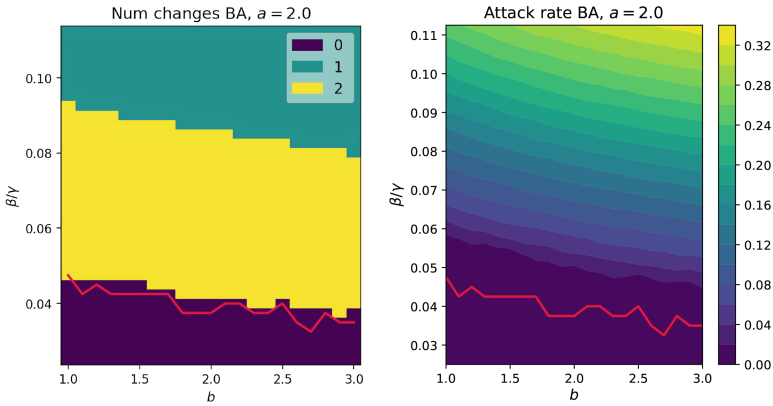


Fig. 6. Results for the BA graph. Left: number of stationary points of $I(t)$. Right: Median attack rate (R_∞). The red lines show the β/γ values corresponding to the peak variability of the attack rate V_{AR}. Other parameters are $p = 0.1$, $\gamma = 0.2$.

is likely that by choosing the epidemic seeds at random, most of them cannot infect many other nodes, and only when a sufficient number of nodes with higher degree are infected, then an epidemic outbreak is possible (see Fig. 7).

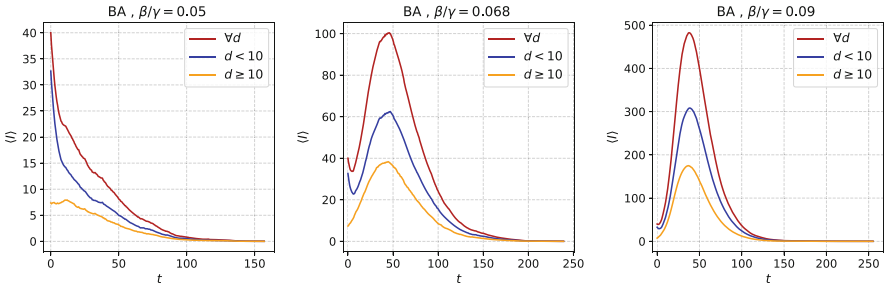


Fig. 7. Average total number of infecteds for $a = 1.0$ and $b = 1.0$ (i.e., ordinary SIR model) on the BA graph, with the average done on a) all nodes ($\forall d$), b) only nodes with degree $d < 10$, c) only nodes with $d \geq 10$. Other parameters are $p = 0.1$, $\gamma = 0.2$.

4 Conclusion

Our study highlights the critical impact of heterogeneity in susceptibility and infectivity on epidemic dynamics, particularly in relation to the epidemic threshold and the emergence of a resurgence zone. By extending the standard SIR model to incorporate a bimodal distribution of these parameters, we demonstrate that the necessary and sufficient conditions for an outbreak shift from a simple consideration of the infected population to the effective infected population, I_{eff}. We remark that the HeSIR model, defined with human behavior in mind, also describes cases where the heterogeneity in transmission originates from biological or other factors.

Our findings reveal that, at least in homogeneous networks, both susceptibility and infectivity must vary for this more complex behavior to emerge, with homophilic networks further widening the resurgence zone. This underscores the importance of considering population heterogeneity when modeling epidemics, as it plays a pivotal role in shaping epidemic trajectories. Disregarding this factor could lead to an underestimation of transmission potential, compromising the intervention in a mitigation scenario. In such a scenario, it would be important to control infections among the non-compliant population, which is the driving force behind the resurgence phenomenon in our findings. This could be accomplished through targeted testing, or by encouraging non-compliant individuals to adhere to safety guidelines. On the other hand, heterogenous structures create the conditions for the resurgence zone to emerge even for the standard SIR model. This result indicates that nodes with highly heterogeneous degree should also be closely monitored.

The HeSIR model presented in this work could be extended in the future to include a higher diversity in susceptibility and infectivity. We also leave to future work the comparison of the outcomes of our model against collected data on epidemic trajectories and the further investigation of the effect of degree heterogeneity.

Acknowledgements. All authors acknowledge support from the project "CODE – Coupling Opinion Dynamics with Epidemics", funded under PNRR Mission 4 "Education and Research" - Component C2 - Investment 1.1 - Next Generation EU "Fund for National Research Program and Projects of Significant National Interest" PRIN 2022 PNRR, grant code P2022AKRZ9, CUP B53D23026080001.

References

1. Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. Rev. Mod. Phys. **74**(1), 47–97 (2002). https://doi.org/10.1103/RevModPhys.74.47. https://link.aps.org/doi/10.1103/RevModPhys.74.47
2. Beca-Martínez, M.T., et al.: Compliance with the main preventive measures of COVID-19 in Spa in: the role of knowledge, attitudes, practices, and risk perception. Transbound. Emerg. Dis. **69**(4), e871–e882 (2022). https://doi.org/10.1111/tbed.14364. https://onlinelibrary.wiley.com/doi/abs/10.1111/tbed.14364
3. Bonaccorsi, S., Ottaviano, S.: Epidemics on networks with heterogeneous population and stochastic infection rates. Math. Biosci. **279**, 43–52 (2016)
4. DeVerna, M.R., Pierri, F., Ahn, Y.Y., Fortunato, S., Flammini, A., Menczer, F.: Modeling the amplification of epidemic spread by misinformed populations. arXiv preprint arXiv:2402.11351 (2024)
5. Funk, S., Salathé, M., Jansen, V.A.: Modelling the influence of human behaviour on the spread of infectious diseases: a review. J. R. Soc. Interface **7**(50), 1247–1256 (2010)
6. Gibson, M.A., Bruck, J.: Efficient exact stochastic simulation of chemical systems with many species and many channels. J. Phys. Chem. A **104**(9), 1876–1889 (2000). https://doi.org/10.1021/jp993732q
7. Miller, J.C.: Epidemic size and probability in populations with heterogeneous infectivity and susceptibility. Phys. Rev. E, Stat. Nonlinear Soft Matter Phys. **76**(1), 010101 (2007). https://doi.org/10.1103/PhysRevE.76.010101
8. Ming, R.X., Liu, J., Cheung, W.K., Wan, X.: Stochastic modelling of infectious diseases for heterogeneous populations. Infect. Dis. Poverty **5**, 1–11 (2016)
9. Mitze, T., Kosfeld, R., Rode, J., Wälde, K.: Face masks considerably reduce COVID-19 cases in Germany. Proc. Natl. Acad. Sci. **117**(51), 32,293–32,301 (2020). https://doi.org/10.1073/pnas.2015954117. https://www.pnas.org/doi/full/10.1073/pnas.2015954117
10. Neri, F.M., Pérez-Reche, F.J., Taraskin, S.N., Gilligan, C.A.: Heterogeneity in susceptible-infected-removed (SIR) epidemics on lattices. J. R. Soc. Interface **8**(55), 201–209 (2011)
11. Qu, B., Wang, H.: SIS epidemic spreading with heterogeneous infection rates. IEEE Trans. Netw. Sci. Eng. **4**(3), 177–186 (2017)
12. Rodrigues, P., Margheri, A., Rebelo, C., Gomes, M.G.M.: Heterogeneity in susceptibility to infection can explain high reinfection rates. J. Theor. Biol. **259**(2), 280–290 (2009)
13. Vilone, D., Vriens, E., Andrighetto, G.: The effect of heterogeneous distributions of social norms on the spread of infectious diseases. J. Phys. Complexity **5**(2), 025012 (2024)
14. Volz, E.: SIR dynamics in random networks with heterogeneous connectivity. J. Math. Biol. **56**, 293–310 (2008)

The Firefighter Game with State-Varying Cost Functions

Ethan Hunter and Jessica Enright(✉)

School of Computing Science, University of Glasgow, Glasgow, UK
jessica.enright@glasgow.ac.uk

Abstract. The Firefighter Problem is a single-player game in which a fire (modelling a rumour, disease, or other contagion) spreads through a graph and we are tasked with defending nodes to save some target number of them. This problem is formally computationally hard in general, but tractable in some restricted cases.

We introduce a Firefighter variant called The Cost Function Firefighter Problem, in which each node has a cost to defend that can vary depending on time and the burning state of the graph - for example, as a function of proximity to fire (which we call *threat*). We show that The Cost Function Firefighter Problem is formally computationally hard, even on a class of trees where classic firefighter is tractable. We complement our theoretical results with an empirical investigation of the performance of threat and degree-based heuristics when our cost functions are threat-based or uniformly random, finding that the relative performance of heuristics is dependent on the class of random graph the process is run on.

Keywords: algorithm · contagion · firefighting

1 Introduction

The Firefighter Problem (Fire) is a deterministic, discrete-time game proposed by Hartnell in [12] to model spread of a fire or contagion in a graph. An instance of Fire consists of an integer k, a graph, and a root node, indicating the initially burning node. In each turn of a game of Fire, nodes may catch fire to become *burning* or be selected to become *defended*. Once burning or defended, a node remains so for the rest of the game. The objective is typically to save the maximum number of nodes from burning.

Many variants of Fire have been studied [8]. Motivated by a desire to include population heterogeneity to this game, we present a new variant called The Cost Function Firefighter Problem (Cost-Fire). The key feature of this new problem is the use of *cost functions* to assign costs to defend to each node that (importantly!) can depend on time and the state of burning in the graph - thus introducing a dynamical aspect. The firefighter is assigned a budget that they may spend on each turn to defend nodes, and may only defend a set of nodes whose costs sum to at most the budget at that time.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 76–86, 2025.
https://doi.org/10.1007/978-3-031-82439-5_7

More formally, a decision-problem instance of Cost-Fire is a graph $G = (V, E)$ with a root r, a computable cost-function c that assigns a natural-number cost to each node given a game state (in terms of burning) and time step, a natural number budget b, and a natural number target k. The question is then whether there exists a sequence of defences that obeys the budget at each time step but saves at least k nodes by the end of the game. An optimisation version of this problem would instead ask the maximum number of nodes that could be saved with a given budget. For computational complexity purposes, we consider the size of the graph, the budget, and the target to be part of the instance size, and assume an oracle for c.

Cost-Fire has been designed with a particular interest in modelling vaccine hesitancy. A survey by Pourrazavi *et al.* considered 91 articles reporting COVID-19 vaccine uptake data, across several countries and programs, and reported an average hesitancy rate of 29.72% [15]. Cost functions can be designed using such data to approximate realistic individual inclination towards being defended.

In the remainder of this paper we describe related work on Fire in Sect. 2, give hardness and algorithmic results in Sect. 3, describe empirical work on heuristics for this problem on random graphs in 4 and conclude in Sect. 5.

2 Related Work

For a review of work on Fire, we direct the reader to a classic review by Finbow and MacGillivray [8]. Here, we mention work that is related to our new variant. Finbow *et al.* used a reduction from a variant of the Boolean Satisfiability Problem (SAT) to show that Fire is NP-hard, even when restricted to trees [8]. We rely on this result for reductions.

Theorem 1. Fire *is NP-hard even when restricted to trees of maximum degree three, but solvable in time polynomial in graph size on such graphs if the root has degree at most two [7].*

There are several graph classes on which Fire can be solved in time polynomial in the size of the input e.g. when the root node is adjacent to every other node in the input graph, when degree is at most three and the root has degree two or less [7], interval graphs, split graphs, permutation graphs, and graphs free of length-k paths for fixed k [9]. We will argue that at least some similar tractability results hold for some types of cost functions for Cost-Fire.

In a game of Cost-Fire (our cost function variant), the defender can select up to some specified *budget* $b \in \mathbb{N}$ nodes to defend per turn. The variant of Fire with a defence budget is called the Budget Firefighter problem - see work by Bazgan *et al.* [1] and Chlebíková and Chopin [3]. Budget Firefighter is NP-complete, even for instances restricted to rooted graphs (G, r) with budget b when the maximum degree of G is $b+2$ and r has maximum degree [5]. Intuitively, NP-hardness follows immediately by reduction from Fire by considering $b = 1$.

Allowing cost functions to vary in time in Cost-Fire introduces *temporality* to the problem. To our knowledge, the only other Fire variant that includes

explicit temporality is the Temporal Firefighter problem (Temp-Fire), which is Fire played on a temporal graph [11] (a graph in which each edge exists only at specified times). Temp-Fire is distinct from Cost-Fire in the latter taking a static (non-temporal) graph and introducing temporality by varying defence costs over time.

Work has been done on the weighted Firefighter problem, in which each node is assigned a static 'weight' value and the aim is to maximise the weights of the saved nodes - see Duffy and MacGillivray [6]. This may sound similar to Cost-Fire, however, the weights in weighted firefighter do not represent costs to defend. In the weighted firefighter problem each node costs the same to defend but offers a 'value,' which we wish to maximise in our saved nodes. In contrast, in Cost-Fire, we aim to maximise the *number of nodes saved,* regardless of their cost.

We also note that limited work has been done on empirical evaluation of heuristic approaches on Fire and variants. To our knowledge, the only such work for the original Fire problem is by García-Martínez et al. [10], who report on the application of heuristics related to degree and proximity to fire to random graph classes - this has inspired our work in Sect. 4.

3 Complexity and Algorithmic Results

Throughout this section we use standard notions in computational complexity, including NP-hardness and associated reduction proof techniques. For the unfamiliar interested reader, we recommend the introductory text by Cormen et al. [4]. We first show hardness of Cost-Fire both straightforwardly from Fire and on a class of graphs where Fire is tractable, and then give a simple case in which Cost-Fire is tractable.

Cost-Fire is a clear generalisation of Fire, so the following is not difficult to justify.

Theorem 2. Cost-Fire *is NP-hard.*

Intuitively, we reduce from Fire by defining an instance of Cost-Fire with budget one and cost function that returns one for every node at every time.

3.1 Hardness

To begin mapping the hardness of Cost-Fire in the landscape of Fire and variants, we define a class of graph that we call *Sea Fans* on which Cost-Fire remains hard but Fire can be solved efficiently. We show an example of a Sea Fan in Fig. 1.

Definition 1 (Sea Fan). *An $(f, \ell)-$Sea Fan (F, r) is a rooted graph defined as follows. The graph $F \setminus \{r\}$ is a forest of f isomorphic trees called fronds, each composed of a path of length ℓ and two nodes adjacent to the terminal node not previously adjacent to r.*

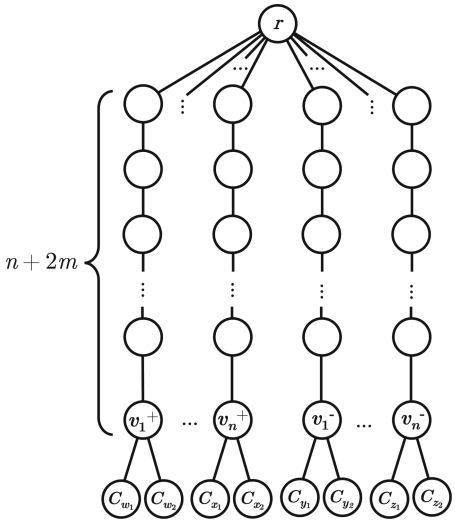


Fig. 1. Construction of a $(2n, n+2m)$-Sea Fan graph, with nodes labelled as per proof of Theorem 3

Theorem 3. Cost-Fire *remains NP-hard even on* $(f, \ell)-$*Sea Fan graphs with temporal cost functions when the fire starts at* r.

Proof. We reduce from the 2N2P-SAT problem, a variant of 3-SAT in which every variable appears precisely twice as a positive literal and twice as a negative literal in the conjunctive normal form of a logical expression in which every clause contains three literals. Observe that a satisfiable expression of this form is composed of $m = 4n/3$ clauses, and without loss of generality we can assume that $n \mid 3$. Berman *et al.* , who refer to this problem as the $(3, 2B)$-SAT problem, show this is NP-complete [2].

To show that Cost-Fire is NP-hard, we give a polynomial-time reduction from an arbitrary instance of 2N2P-SAT such that the answer to the Cost-Fire instance is 'Yes' if and only if the 2N2P-SAT instance is satisfiable.

Take as given a 2N2P-SAT instance ϕ of n variables in m clauses for some $3 \mid n$. Let $\mathcal{L}$ and $\mathcal{C}$ be lexicographic orderings of the literals and clauses of ϕ respectively. We create an instance of Cost-Fire on a $(2n, n+2m)$-Sea Fan (F, r), as in Fig. 1, as follows: For each variable v_i, the terminal node of path P_i not adjacent to r in F is labelled v_i^+ and the analogous node of P_{n-i} is labelled v_i^-, corresponding to the positive and negative literals of v_i respectively. For each of these literal-labelled nodes, label its two neighbours of degree one C_j and C_k, where j and k are the indices of the clauses in $\mathcal{C}$ in which the literal appears, if such clauses exist. If not, remove the neighbouring nodes.

We define a cost function $c : V \times \mathbb{N} \to \mathbb{N}$ that assigns all nodes a cost of 2 at all times, except: nodes v_i^+ and v_i^- each cost 1 at time i for $1 \leq i \leq n$, and all three nodes labelled C_j cost 1 at times $n+2j-1$ and $n+2j$ for $1 < j \leq 2m$.

For the remaining input variables, let $b = 1$ and $k = n + 3m$. Intuitively, we can defend one literal-labelled node for each variable and two clause-labelled nodes for each clause. For all three clause-variables C_j to be saved, at least one node labelled with a literal in C_j must have been defended - this will correspond to a satisfied literal in the 2N2P-SAT instance.

Given the satisfying 2N2P-SAT instance ϕ, we deploy the following strategy for Cost-Fire on the constructed Sea Fan to save at least $k = n + 3m$ nodes. At each time $1 \leq i \leq n$, defend the node labelled v_i^+ if v_i is true in ϕ, v_i^- otherwise. As ϕ is satisfying, at least one of the three clause-labelled nodes are now defended for each clause. At each pair of times $j, j+1$ from $(n+1, n+2)$ to $(3n-1, 3n)$, let $k = j - n$ for simplicity and defend the nodes labelled C_k and C_{k+1} in $\mathcal{C}$. By saving the remaining two clause-labelled nodes for each clause, we save all three nodes associated with each clause in ϕ, and overall save at least $k = n + 3m$ nodes.

We now argue that, given an instance of Cost-Fire on the constructed Sea Fan on which we can save at least $n+3m$ nodes, we obtain a satisfying assignment for the corresponding instance of 2N2P-SAT. Observe that only literal- and clause-labelled nodes can be saved, and only n literal nodes can be saved - either v_i^+ or v_i^- for each variable v_i. Assume we have a 'Yes' instance of Cost-Fire on the constructed Sea Fan graph. Then, at least $3m$ clause nodes must be saved; for each clause, there are three nodes and only two can be directly defended. Thus, one must have been saved by the defence of a parent literal-labelled node. Therefore, there is at least one literal-labelled node for each clause that is directly defended, and interpreting these as truth values gives a satisfying assignment.□

Theorem 4. Fire *is decidable in polynomial time on* (f, ℓ)*-Sea Fan graphs when the fire starts at the root node.*

Proof. There exists an optimal strategy that always defends adjacent to burning nodes for instances of Fire on trees [14]. On a Sea Fan graph, if we defend adjacent to fire on a frond, fire is contained in this frond. Because of the symmetry of the fronds, all defence choices adjacent to the fire on a frond will give an equivalent game state. Thus, there is an optimal defence that defends an open node adjacent to a burning node at each time, unless no such node exists in which case the process is over.

Then an optimal strategy defends a node on a frond at distance one from the root at time one, at distance two on a frond at time two, etc. - at each time t the strategy will defend a node at distance t until the end of the process. This can be simulated in polynomial time, and thus Fire can be decided efficiently on an (f, ℓ)-Sea Fan graphs when the fire starts at the root node.

We depict an example of this strategy in Fig. 2.

3.2 Tractability

Fire is known to be solvable in time polynomial in graph size on graphs for which the root is connected to every other node (i.e. is a universal node). Intuitively,

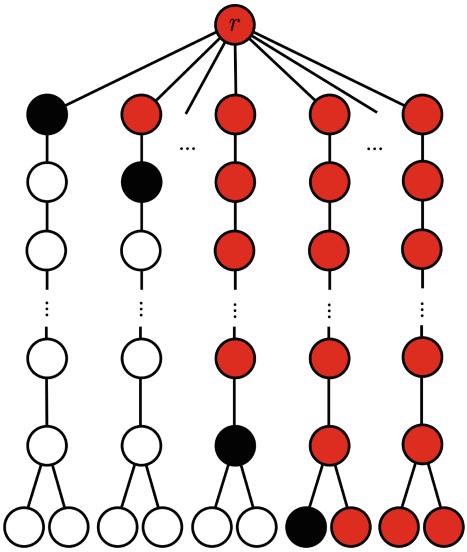


Fig. 2. Depiction of strategy used in proof of Theorem 4, where: unfilled nodes are open, red nodes are burning and filled-in nodes are defended.

Cost-Fire on complete graphs on n nodes is solvable in $\mathcal{O}(n \ln n)$ time as we need only sort the nodes from least to greatest cost and defend in this order in the first turn until the budget is spent. More interestingly, and inspired by Fomin et al. [9], we show that we can resolve Cost-Fire efficiently on graphs with bounded path length.

We say a graph is P_ℓ*-free* if it contains paths of maximum length $\ell - 1$ (this class is usually called P_k-free, but k is already a variable in Fire). We consider these graphs because Fire on P_ℓ-free graphs on n nodes is known to be solvable in $\mathcal{O}(n^\ell)$ time [9].

Theorem 5. *Instances of* Cost-Fire *with budget* $b \in \mathbb{N}$ *on* P_ℓ*-free graphs with* n *nodes and* m *edges can be solved in time* $\mathcal{O}((n+m)n^{b(\ell-2)})$.

Proof. Consider an instance of Cost-Fire on: a rooted graph (G, r) on n nodes and m edges; an integer k; a budget $b \in \mathbb{N}$; and an arbitrary cost function. Observe that any strategy defends at most $\ell - 2$ sets of nodes, as this is bounded by the length of the longest path. Hence, we list the sequences of at most $b(\ell-2)$ nodes in G - there are $\mathcal{O}(n^{b(\ell-2)})$ of them - and partition each sequence into cells of size at most b with respect to cost to defend. We play through each strategy, returning the first valid strategy that saves at least k nodes in $\mathcal{O}(n+m)$ time using an appropriate graph search. This algorithm runs in $\mathcal{O}((n+m)n^{(b\ell-2)})$ time. □

Corollary 1. Cost-Fire *on* P_ℓ*-free graphs is resolvable in polynomial time when* ℓ *and the budget are held constant.*

4 Empirical Heuristic Approaches

We are motivated by the formal hardness of Cost-Fire and an interest in behaviour of the game in practice to investigate the performance of a number of heuristics on different random graphs and cost functions.

We therefore report on the number of nodes saved by several heuristic approaches, over several cost functions on random regular graphs and Barabási-Albert graphs. We have selected these for interest, and the fact that they are widely known; we propose a broader investigation and comparison to optimal solutions as future work. The parameters of these preliminary experiments

4.1 Cost Functions

Here, we report on two styles of cost function with some variation between them: uniform random costs, and *threat-based* costs (i.e. based on proximity to fire) in which the cost of defending a node is lower if it is closer to the fire. All costs are natural numbers. In particular, we implement:

- a uniform random cost varying between 1 and 5
- a threat-based cost equal to the graph distance to the closest burning node
- a threat-based cost as above but with an added stochastic factor uniformly distributed between $\{-h, h\}$ for $h \in \{1, 3\}$.

For verification purposes, we also implemented and tested using a uniform unit cost - this allowed comparison to classic firefighter instance benchmarks - but we do not report these here.

All experiments are on graphs of 50 nodes, with a defence budget of five per turn. We ran 1000 replicates of each trial, regenerating the costs and graph at each trial. We suggest investigate on a wider range of larger graphs as future work.

4.2 Heuristics

Inspired partially by previous firefighter work in [10], we implemented heuristics based on degree, cost, and threat. Note therefore that in our implementation threat can influence defender choice via both the cost function and the heuristic function. The degree heuristic prefers nodes of high degree, the threat heuristic nodes that are closer to the fire, the cost heuristic prefers nodes that have lower cost. We consider versions of these that sort first by their primary heuristic preference, and then tie-break based on a second. Overall, we report results on heuristics prioritising by:

- degree alone,
- cost alone,
- degree with ties broken by cost, and
- threat with ties broken by cost.

Other combinations and options are possible - notably heuristics that include number of descendants. We suggest these as areas of future work.

4.3 Graph Classes

We have limited our initial investigation to random regular graphs and Barabási-Albert (BA) graphs, with densities chosen for interest-level of results: too sparse and we see little fire spread, too dense and we see almost all nodes burn regardless of defence strategy. Our random regular graphs have degree six, and our BA graphs add six edges when each node is added - we are aware these are not equivalent numbers of edges.

4.4 Results

Unsurprisingly, we find that our different heuristics perform differently on BA an random regular graphs. In both cases, the heuristics performed worst on uniform random costs, and better when costs were threat-related.

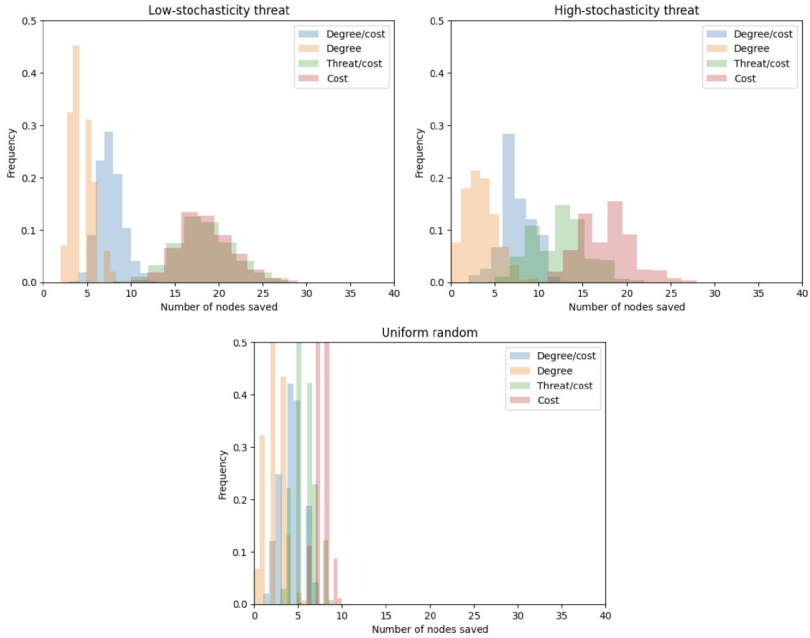


Fig. 3. Frequency plots of trials by nodes saved on BA graphs of 50 nodes when heuristically solving the Cost-Fire game, using four different heuristic prioritisation as in the legends. Costs by low-stochasticity threat on the top left, high-stochasticity threat on the top right, and uniformly random on the bottom.

In Fig. 3 we see that on our BA graphs the threat (with ties broken by cost) and cost-based heuristics perform similarly in the low-stochasticity setting - this is as we would expect when the costs are essentially the same as threat. In the high-stochasticity threat setting these two heuristics separate, with cost-based

performing better. In both the low- and high-stochasticity settings the degree-based heurstics performed worse, and all heuristics performed badly when costs were set uniformly at random at each step of the game.

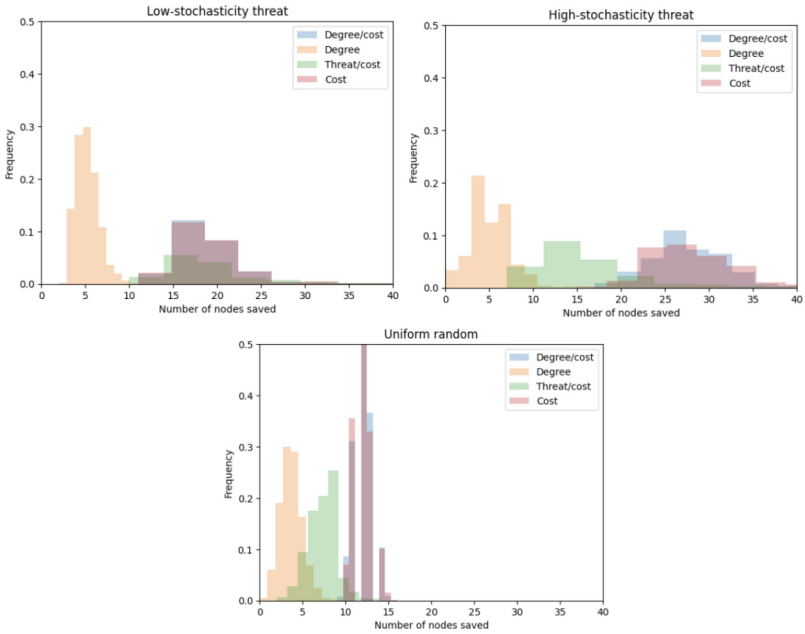


Fig. 4. Frequency plots of trials by nodes saved on random 6-regular graphs when heuristically solving the Cost-Fire game, using four different heuristic prioritisation as in the legends. Costs by low-stochasticity threat on the top left, high-stochasticity threat on the top right, and uniformly random on the bottom.

In Fig. 4 we see that in our random 6-regular graphs the threat, degree (ties broken by cost), and cost-based heuristics perform similarly in the low-stochasticity setting: unsurprising, as all degrees are equal and the cost and threat are similar - so here thees three heuristics essentially make the same priority choices. In the high-stochasticity setting degree/cost and cost outperform the threat/cost heuristic - again, here the degree/cost heuristic will be making the same choices as the cost-based one. In both low and high-stochasticity settings the degree-based heuristic is essentially choosing with no information, as all degrees are the same. Unlike for BA graphs, in Fig. 4 we see that in the case of uniformly random costs there is a difference between performance of heuristics, with cost and degree/cost heuristics performing the best.

It is notable that for random regular graphs the overall performance of the better-performing heuristics is stronger than in the high-stochasticity threat setting than the low-stochasticity threat setting: the increased noise is increasing performance. However, it is clear from the fact that the results are worst in the

uniform random costs setting that some threat signal is valuable. We suggest a further investigation of this as future work.

5 Conclusion

We have introduced a new variant of the Firefighter problem that includes dynamic cost functions for nodes, allowing us to defense cost that varies in time or in response to burning in the graph. We have given basic theoretical results of both hardness and tractability, as well as heuristic performance results on random graphs.

We suggest a number of areas for possible future work: algorithmically, it might be fruitful to investigate computational complexity on graph classes such as interval and permutation graphs. Empirically, we suggest investigation including exact solutions via integer linear programming or similar, as well as simulation of additional more complex heuristics. We gave particular attention to threat-focused heuristics, which could be considered analytically using existing results for shortest path distributions [13]. For preliminary experiments, we selected only limited random graph classes on small numbers of nodes; we suggest simulation on a broader range of random graphs, as well as graphs derived from real-world data.

References

1. Bazgan, C., Chopin, M., Ries, B.: The firefighter problem with more than one firefighter on trees. Disc. Appl. Math. **161**(7–8), 899–908 (2013)
2. Berman, P., Karpinski, M., Scott, A.: Approximation hardness of short symmetric instances of MAX-3SAT. Electronic Colloquium on Computational Complexity (ECCC) (2003)
3. Chlebíková, J., Chopin, M.: The firefighter problem: further steps in understanding its complexity. Theor. Comput. Sci. **676**, 42–51 (2017)
4. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. MIT press, Cambridge (2022)
5. Duffy, C.: A collection of algorithmic and complexity results for variants of the firefighter problem. Master's thesis, University of Waterloo (2011)
6. Duffy, C., MacGillivray, G.: An analysis of the weighted firefighter problem. J. Comb. Math. Comb. Comput. **94**, 167–175 (2015)
7. Finbow, S., King, A., MacGillivray, G., Rizzi, R.: The firefighter problem for graphs of maximum degree three. Disc. Math. **307**(16), 2094–2105 (2007)
8. Finbow, S., MacGillivray, G.: The firefighter problem: a survey of results, directions and questions. Aust. J. Comb. **43** (2009)
9. Fomin, F.V., Heggernes, P., van Leeuwen, E.J.: The firefighter problem on graph classes. Theor. Comput. Sci. **613**, 38–50 (2016). https://doi.org/10.1016/j.tcs.2015.11.024. https://www.sciencedirect.com/science/article/pii/S0304397515010853
10. García-Martínez, C., Blum, C., Rodriguez, F.J., Lozano, M.: The firefighter problem: empirical results on random graphs. Comput. Oper. Res. **60**, 55–66 (2015)

11. Hand, S.D., Enright, J., Meeks, K.: Making life more confusing for firefighters. In: P. Fraigniaud, Y. Uno (eds.) 11th International Conference on Fun with Algorithms (FUN 2022), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 226, pp. 15:1–15:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl (2022). https://doi.org/10.4230/LIPIcs.FUN.2022.15. https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.FUN.2022.15
12. Hartnell, B.L.: Firefighter! an application of domination. In: 25th Manitoba Conference on Combinatorial Mathematics and Computing. University of Manitoba in Winnipeg, Canada (1995)
13. Katzav, E., et al.: Analytical results for the distribution of shortest path lengths in random networks. EPL (Europhys. Lett.) **111**(2), 26,006 (2015). https://doi.org/10.1209/0295-5075/111/26006
14. MacGillivray, G., Wang, P.: On the firefighter problem. J. Comb. Math. Comb. Comput. **47**, 83–96 (2003)
15. Pourrazavi, S., Fathifar, Z., Sharma, M., Allahverdipour, H.: Covid-19 vaccine hesitancy: a systematic review of cognitive determinants. Health Promot. Perspect. **13**(1), 21–35 (2023). https://doi.org/10.34172/hpp.2023.03

Optimal Intervention Campaign to Combat Anti-vaccine Social Contagion and Contain Epidemic Spread

Sarah Alahmadi[1](✉), Rebecca Hoyle[2], Michael Head[3], and Markus Brede[1]

[1] School of Electronics and Computer Science, University of Southampton, Southampton, UK
s.h.alahmadi@soton.ac.uk
[2] School of Mathematical Sciences, University of Southampton, Southampton, UK
[3] Clinical Informatics Research Unit, Faculty of Medicine, University of Southampton, Southampton, UK

Abstract. Vaccine misinformation is a significant driver of vaccine hesitancy, spreading within social networks as anti-vaccine social contagion. This potentially causes larger disease outbreaks due to the clustering of unprotected individuals influenced by this contagion, who may then choose to remain unvaccinated. In this study, through a coupled agent-based model that integrates vaccine opinions and disease diffusion processes, we design an optimal counter-campaign that spreads positive vaccine information to counteract vaccine misinformation and ultimately suppress the spread of an epidemic. We demonstrate that targeting individuals with anti-vaccine neighbors in their social networks can effectively contain the spread of negative influence when interventions are implemented at early stages. Conversely, once anti-vaccine opinion adopters begin forming larger clusters within the network, shielding bridging regions between clusters becomes crucial in restricting the growth of anti-vaccine communities and thereby controlling the spread of epidemics.

Keywords: epidemic dynamics · anti-vaccine opinion · vaccine hesitancy · complex contagion · complex networks

1 Introduction

Vaccine hesitancy poses a significant obstacle to the vaccination process and the control of vaccine-preventable diseases spread. According to WHO, vaccine hesitancy is one of the top ten threats to public health [1]. One key driver of this hesitancy is the spread of vaccine misinformation in social networks [2–4,6]. One real-world instance of its adverse effects occurred in 2019, when a decline in measles vaccine coverage led to a significant increase in measles cases [2]. Similarly, misinformation has negatively impacted COVID-19 vaccine uptake [3]

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 87–99, 2025.
https://doi.org/10.1007/978-3-031-82439-5_8

and has decreased the likelihood of individuals engaging in preventive behaviors, including accepting Ebola vaccines during the Ebola outbreak [7].

Studying the impact of information transmission on the spread of infectious diseases has drawn the attention of numerous researchers through coupled dynamics models [5,8,11,15,16,19,20]. Some studies investigate the problem in non-competitive scenarios where only one type of information is diffused in the network. This diffusion can either trigger personal self-protection by raising individuals' awareness [28] or influence people's willingness to get vaccinated by spreading vaccine misinformation [5,8,20]. While other studies consider the competitive diffusion of both negative and positive information [11,16,29,31]. Despite this considerable attention, there is a paucity of research explicitly exploring the impact of mitigating the diffusion of vaccine hesitancy, embodied as anti-vaccine opinions, on the dynamics of infectious diseases. Designing exogenous behavioral interventions to control disease spread and anti-vaccine influence diffusion remains an active research area with limited studies [11,12,17,31]. The authors in [11] considered an intervention scheme to control disease spread by managing the spread of anti-vaccine opinions, assuming that any agent converted to an anti-vaccine opinion could return to a pro-vaccine state through dedicated influence. Meanwhile, [12] investigates the impact of an intervention campaign aimed at increasing preventive behaviors, particularly by increasing social distance between agents. A recent study proposed more strategic intervention campaigns to control vaccine hesitancy diffusion, but they studied the problem as a single diffusion process and did not consider the effects of these interventions on the disease dynamics [17]. In addition, a recent study that closely resembles our work [31] investigates the impact of different targeting heuristics for placing pro-vaccine seeds on epidemic size. While there is extensive literature on strategic targeting for node immunization in vaccination campaigns [14], as well as within single diffusion processes for misinformation mitigation [13,27,30] and influence maximization [9], there is a notable lack of studies addressing such strategic targeting schemes to mitigate vaccine hesitancy and capture their effects within the complex interplay between opinion propagation and disease transmission. Moreover, considering the spatial distribution and connectivity patterns of vaccine-hesitant individuals within the social network, particularly their clustering across different network regions, are crucial factors that can profoundly impact the spread of the epidemic. Existing studies have highlighted the clustering of anti-vaccine opinions in online social networks [23] and unprotected individuals in spatial networks [24]. Additionally, numerous studies have examined the risk posed by the clustering of unprotected individuals and their impact on achieving herd immunity[11,25,26].

To bridge these gaps, in this study, we present a coupled agent-based model that integrates a model for vaccine-opinion exchange with the process of disease spread. Through this integrated framework, we aim to design an optimal intervention strategy that combats the spread of negative vaccine opinions by spreading positive vaccine information to enhance vaccine uptake. This strategy targets individuals based on the dynamic evolution of anti-vaccine attitudes,

with the overarching goal of counteracting the propagation of anti-vaccine sentiments and ultimately suppressing the spread of an epidemic outbreak. Unlike heuristic-based methods to target agents in one-off settings [31] and in multiple interventions in our previous work [32], here we address the problem as a constrained optimization problem with a limited budget, aiming to minimize the epidemic size through minimizing the number of anti-vaccine opinion adopters in a competitive setting where two campaigns spread competing information—anti-vaccine and pro-vaccine—across the network. Inspired by techniques of continuous influence maximization [9], we propose an adaptive campaign that continuously allocates resources and targets agents, in contrast to one-off targeting approaches commonly employed [e.g., [13,27,31]]. Moreover, considering the significant concern posed by anti-vaccine clusters in the context of vaccine-preventable diseases, that are also evident in our findings, we extend our continuous optimization framework to mitigate the formation of large anti-vaccine communities. Therefore, in this study, we contribute to the literature by not only minimizing the number of negative opinion adopters but also by strategically disrupting the interconnectivity between anti-vaccine communities. We propose an intervention strategy that aims to disrupt the formation of large anti-vaccine clusters by strategically promoting vaccine uptake in critical network regions, thereby impeding the integration of existing isolated communities into larger, interconnected communities.

Our main findings are as follows: reducing the total number of adopters of anti-vaccine opinion can effectively mitigate negative influence and decrease the epidemic size when the intervention takes place early, before widespread adoption occurs. However, in later stages and under scenarios where anti-vaccine sentiments have already spread, a cluster-based approach demonstrates greater efficiency. Furthermore, we demonstrate that targeting agents who can initiate positive counter-contagion is a more potent strategy than protecting the most vulnerable agents from negative influences, yielding better outcomes in misinformation management, vaccine uptake, and epidemic control. Additionally, initiating a positive social contagion in critical network areas plays a pivotal role in preventing the formation of large anti-vaccine clusters, thereby reducing the epidemic size.

The rest of the paper is structured as follows. In Sect. 2, we outline the model used for diffusion processes and the proposed intervention campaign for dynamic allocation. Section 3 presents the primary results concerning optimal dynamic allocation and targeting schemes. Finally, Sect. 4 concludes with a summary and outlines future work.

2 Model Description and Methods

We consider a population of N agents interconnected via a small-world network. Initially, we assume that all agents hold a neutral opinion about vaccines and can adopt either an anti-vaccine opinion, pro-vaccine opinion, or remain neutral. Motivated by the approach of [8], our model then consists of two diffusion stages: vaccine opinion diffusion followed by disease spread. Inspired by

their work, we developed a model for vaccine opinion diffusion that addresses the dual propagation of positive and negative opinions, as investigated in our previous work [32]. We assume that once an individual adopts a pro-vaccine opinion, they opt for immunization, assuming full immunity—especially relevant for childhood diseases, e.g., measles. In contrast, anti-vaccine individuals decline vaccination and do not change their opinion. Similarly, as in [8], we utilized the SIR model for the disease spread stage, originally developed by [18], which is widely used by researchers in epidemic modeling. In addition, we utilize a complex contagion model to capture the spread of vaccine opinions, where agents need multiple exposures to be influenced, reflecting observed patterns in diffusion processes [21,22]. By taking into consideration the correlation between the number of adopters of anti-vaccine opinion and the extent of an epidemic, our goal is to minimize the epidemic size by minimizing the expected number of anti-vaccine opinion adopters. However, predicting the long-term co-evolution of the these dynamics—i.e., vaccine opinions and disease spread—is challenging due to the stochastic nature of such processes. Thus, we propose a one-step-ahead optimization approach to reduce the estimated number of individuals adopting anti-vaccine opinions in subsequent time steps, based on the current state of the social network, i.e., the information about population attitudes toward the vaccine.

To begin, we derive the equations that govern the opinion dynamics. The probability of an agent transitioning from a neutral opinion state to an anti-vaccine opinion state Pr_i^- at time t depends on: (i) social influence which originates from anti- and pro-vaccine neighbors within its social network K_i^- and K_i^+, respectively, each with a probability ω of exerting influence at each time step; (ii) external influence (e.g., media, campaigns) which influences individuals with strengths of μ^- and μ^+, for anti-vaccine and pro-vaccine information at each time step; (iii) the agent's previous exposures to other influence, which might be positive or negative. We label the effective difference between exposures to positive and negative opinions by Δ_i, defined as $\Delta_i = \phi_i^+ - \phi_i^-$, where ϕ_i^+ represents the total number of positive exposures and ϕ_i^- represents the total number of negative exposures experienced by an agent i. An agent needs to be exposed to at least θ more exposures of positive or negative influence in order to adopt the corresponding opinion. We then aim to optimally allocate the positive strength μ_i^+, $i = 1, .., N^0$, where N^0 represents the set of current agents with neutral opinion. We find that the probability of adopting an anti-vaccine opinion is given by:

$$Pr_i^-(X_i \geq Y_i + \theta) = \sum_{a=0}^{K_i^- - (\theta + \Delta_i)} \sum_{j=a+(\theta+\Delta_i)}^{K_i^-} P_i^-\left(X_i = j\right) \cdot P_i^+\left(Y_i = a\right), \quad (1)$$

where $P_i^-\left(X_i = j\right)$ is the probability of obtaining a total of $X_i = j$ negative exposures, and $P_i^+\left(Y_i = a\right)$ is the probability of obtaining a total of $Y_i = a$ positive exposures. We obtain P_i^- as follows: $P_i^-(X_i = x) = \mu_i^- \cdot p_i^-\left(x-1\right) + \left(1 - \mu_i^-\right) \cdot p_i^-\left(x\right)$, where $\mu_i^- = \mu^-$ represents exposure from external influence

and p_i^- represents exposures from social contacts, calculated as follows: $p_i^-(x) = \binom{K_i^-}{x}(\omega)^x \cdot (1-\omega)^{K_i^- - x}$. We find an analogous expression for P_i^+.

Let $\langle \Delta n^- \rangle = \sum_{i \in N^0} Pr_i^-$ be the average number of new anti-vaccine opinion adopters in the next time step $(t+1)$. In the following, our objective is to minimize this function with respect to the positive campaign allocation. In this stage, inspired by techniques discussed in [9], we aim to continuously allocate optimal influence rate μ_i^+ to each target agent i within a total budget $B = \mu^+ \times N$. This problem can be formalized as a constrained optimization problem:

$$\min_{\boldsymbol{\mu}^+} \langle \Delta n^- \rangle_{(t+1)}, \text{ s.t.} \qquad \sum_i \mu_i^+ \leq B, \qquad 0 \leq \mu_i^+ \leq 1. \tag{2}$$

Given the linear nature of the objective function and the associated constraints, we leverage linear programming methods, particularly the simplex method, to derive the optimal solution. The experiment begins with the opinion diffusion stage, which incorporates our intervention campaigns. During this phase, vaccine sentiments spread through the network, influencing individuals' vaccination decisions until all neutral opinions are changed to either positive or negative opinions. In the disease propagation stage, the disease spreads among unvaccinated individuals, initiated by randomly selected anti-vaccine adopters (I_0). The process continues until there are no more infected individuals, and the total number of cases is then recorded as the epidemic size.

3 Results

We begin our analysis by exploring the impact of minimizing the number of anti-vaccine opinion adopters on the extent of the epidemic size. In this campaign, we assume full knowledge regarding vaccine attitudes and historical information about previous exposures an agent has encountered, referred to as (Full_Kno.). We also consider a more realistic scenario where we lack full knowledge about historical exposures, assuming $\phi_i^+ = \phi_i^- = 0$, referred to as (No_Kno.). Additionally, if the allocation has not fully utilized the budget B, we randomly select T nodes and uniformly allocate the remaining budget. We then propose an optimal campaign to restrict the growth of anti-vaccine communities in the network. In all experiments, following a similar configuration used in [8] we consider a small-world network [10] with size $N = 5000$, rewiring probability $p = 0.01$, and average degree $\langle k \rangle = 10$. We use an opinion formation threshold of $\theta = 2$. For SIR parameters we use infection rate $\beta = 0.1$, recovery rate $\gamma = 0.1$, and seed set $I_0 = 1$. In all presented figures, the shaded area and error bars represent the 95% confidence intervals. We use a dynamic random campaign proposed in our previous work [32], referred to as dynRand, as a benchmark. In this approach, campaign resources are uniformly allocated to a randomly chosen target set of size T, with each target receiving an allocation of μ_i^+. This target set is updated at each time step t.

3.1 Minimizing Anti-vaccine Opinion Adopters

To facilitate the analysis, we commence with a simplified campaigning scheme where both the positive and opponent campaigns are initiated simultaneously at $t = 0$. Subsequently, we consider a more realistic scenario involving late interventions, wherein the positive campaign is launched late to counteract an already existing negative propagation. Figure 1 displays the results of this campaign in both the Full_Kno. and No_Kno. settings.

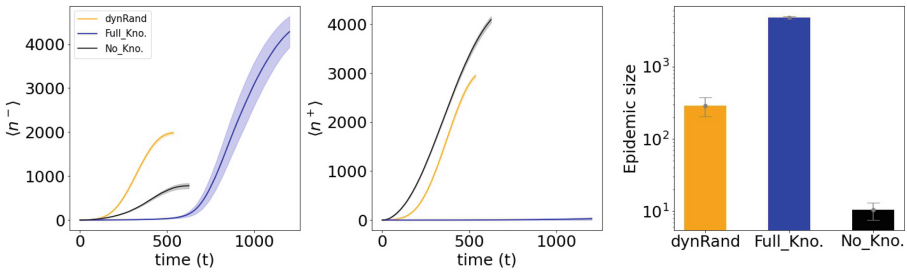


Fig. 1. Opinion diffusion and epidemic size for the optimal campaigns starting at time t = 0 are compared to the dynRand campaign. The left panel represents the evolution of anti-vaccine opinion adopters, the middle panel represents the evolution of pro-vaccine opinion adopters (vaccinated), and the right panel represents the corresponding epidemic size. Results are the average of 50 realizations with $\mu^+ = \mu^- = 0.001$, $\omega = 0.006$, and $T = 5$.

As depicted in Fig. 1, the (Full_Kno.) campaign effectively delays the spread of anti-vaccine attitudes, as illustrated by the blue line in the left panel. This results in a significantly slower development of anti-vaccine opinion adopters over an extended time period. In Fig. 2, we provide snapshots of the network configuration at different time steps during the opinion diffusion stage to analyze the targeting scheme. The figure shows the neighborhood structure (number of anti- and pro-vaccine neighbors) for all candidate agents (neutrals) and identifies the campaign targets. The campaign primarily targets nodes with the highest number of anti-vaccine neighbors (x-axis) in both settings. In the full knowledge setting, it further prioritizes nodes previously influenced by the opponent—agents with $\Delta = -1$ among these targets—as they are the most likely to convert to a negative state in the next time step.

As the campaign targets nodes with a higher number of anti-vaccine neighbors, it successfully shields the most vulnerable agents from negative influence, preventing them from transitioning from a neutral state to an anti-vaccine opinion state. Consequently, the campaign performs well initially because the network configuration helps contain the spread of negative adopters. At this stage, most nodes have fewer negative neighbors and a larger number of neutral neighbors. By protecting those nodes from converting to a negative state, the campaign also prevents the initiation of a negative social contagion and indirectly

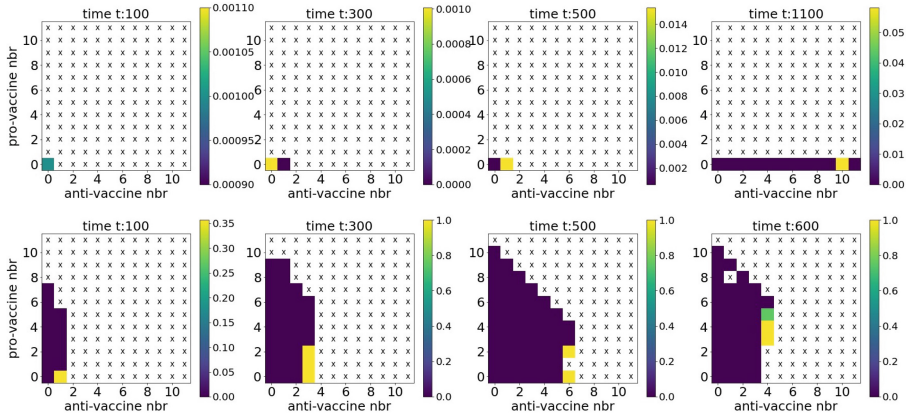


Fig. 2. The targeting scheme for the optimal campaign is illustrated with full knowledge (top panel) and without full knowledge (bottom panel). These panels depict the neighborhood structure at a single time step t during the opinion diffusion stage, showing the number of anti-vaccine and pro-vaccine neighbors for all candidate agents. Each cell indicates the average allocation for nodes with x anti-vaccine neighbors and y pro-vaccine neighbors. An 'x' indicates that no agent exists for that neighborhood pattern. Targeted individuals are those with an allocation not equal to zero (non-purple color). (Color figure online)

protects those neutral neighbors, thus preventing the emergence and growth of anti-vaccine clusters.

However, the campaign falls short in halting the overall negative diffusion in the network, resulting in a significant increase in the number of anti-vaccine adopters. Consequently, it fails to curtail the spread of the epidemic, as shown by the blue bar in the right panel of Fig. 1. This increase is due to two factors: first, the limited budget of the positive campaign, and second, the mechanism of the campaign in this setting. The campaign in this full knowledge setting provides significant protection for the most vulnerable individuals against negative influence. It focuses purely on protection and does not contribute to fostering positive collective behavior. The positive exposures given are not sufficient to convince agents to get vaccinated. Due to complex contagion, agents need multiple exposures to be convinced to vaccinate, and the campaign avoids agents who have already been targeted and protected from anti-vaccine influence. Consequently, as many negative adopters emerge, this negative behavior begins to spread. Recall that the negative influence is bolstered by an external negative campaign that continues to influence the population randomly. Additionally, with limited resources, we fail to protect all agents in the system and, due to the targeting criteria, cannot initiate positive contagion in the network. This leads to a rapid diffusion of anti-vaccine adopters. Therefore, the campaign fails to combat the negative propagation effectively over time.

The targeting criteria in this campaign are primarily determined by previous exposures to the opponent's influences, in addition to the neighborhood structure. This raises the question: What if we do not consider previous exposures? Would this enhance the promotion of positive collective behavior? To explore this, we relax the assumption of prior knowledge about previous exposures, which also aligns more closely with a realistic scenario. This change results in resource allocation being determined solely by the neighborhood structure of the nodes, without considering past exposures. Consequently, we observe that the campaign performs much better, as seen in Fig. 1 (black lines for left and middle panels, and black bar for right panel).

The campaign in the no-knowledge setting effectively fosters positive influence propagation, see black line in the middle panel of Fig. 1. We can see that the number of vaccinated individuals significantly increases in this scenario. This increase occurs because agents can be targeted multiple times, unlike in the previous setting where the campaign avoided previously targeted agents. Thus, the repeated exposures are sufficient to convince them to get vaccinated. In addition, by examining the targeting scheme, see the bottom panels of Fig. 2, the campaign still targets those with a higher number of anti-vaccine neighbors. However, as depicted in the figure, we can see from the network configuration that there is growth in the number of pro-vaccine neighbors (y-axis) compared to the top panels. This indicates that this campaign not only protects vulnerable individuals against negative influence but also effectively promotes vaccine uptake in regions close to the negative areas in the network. This results in the initiation of positive behavior within negative areas, aiding in containing the negative spread alongside providing protection, at early times ($t < 500$), before the clustering of anti-vaccine opinion adopters in the network. This clustering is depicted by the presence of agents with a high number of anti-vaccine neighbors, as illustrated in the network configuration in Fig. 2. See the network configuration at time $t = 500$ as an example, where there are agents with 6 anti-vaccine neighbors indicating the onset of clustering. Ultimately this campaign leads to a significant reduction in the size of the epidemic, see black bar in in the right panel of Fig. 1.

Until now we have assumed a simple scenario where both campaigns, i.e., opponent and positive, start simultaneously at time $t = 0$. Next, we evaluate the efficiency of the campaign in a more realistic scenario by applying it to a network where clustering and negative attitudes are already present. Can we contain this spread and mitigate the growth of the anti-vaccine communities? What will the shape of the optimal campaigns be in mitigating the spread of anti-vaccine influence? Here we consider a scenario where anti-vaccine sentiments are already prevalent in the network and we intervene at later stages. Figure 3 shows the obtained final epidemic size of the optimal campaign with late interventions in both settings compared to the benchmark campaign. The figure illustrates the results of interventions occurring at different times: $t = 50$ (left panel) with approximately 0.001 of the population having adopted an anti-vaccine opinion, $t = 100$ (middle panel) with approximately 0.01 of the population having

adopted an anti-vaccine opinion, and $t = 200$ (right panel) with approximately 0.1 of the population having adopted an anti-vaccine opinion.

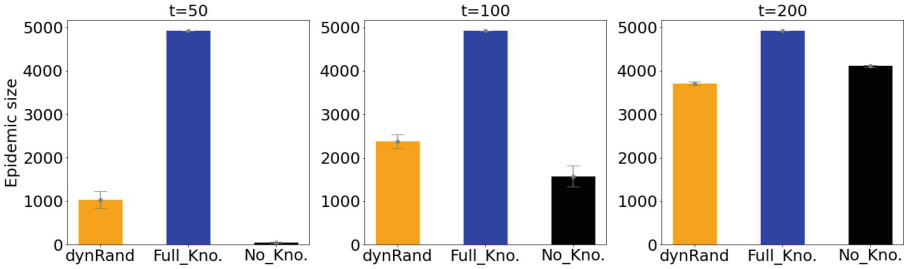


Fig. 3. Epidemic size for different optimal campaigns starting at time t are compared to the dynRand campaign. Results are the average of 50 simulations with Budget $B = 5$ and parameters $\mu^+ = \mu^- = 0.001$, $\omega = 0.006$, and $T = 5$.

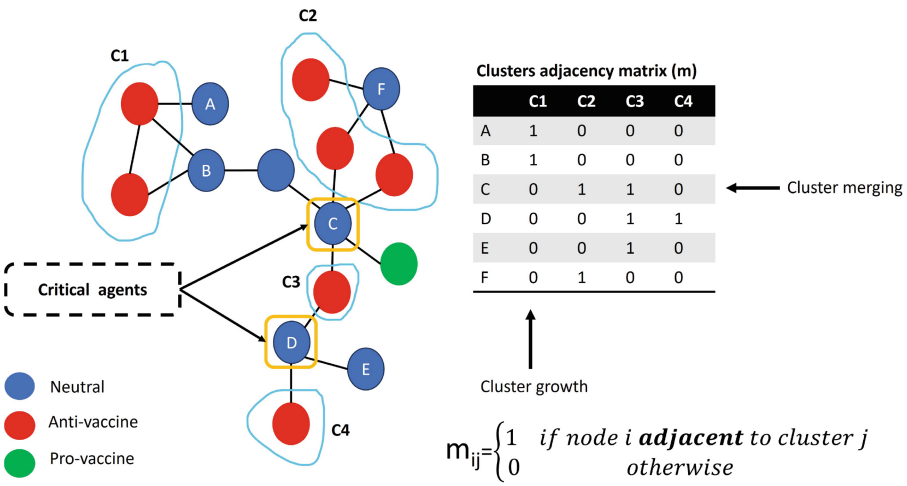
A delayed intervention, occurring at $t = 200$ in the right panel of Fig. 3, results in an earlier onset of broader social contagion and clustering among anti-vaccine opinion adopters, leading to the formation of larger anti-vaccine communities. Consequently, this results in poorer performance of the optimal campaigns in containing the growth of the anti-vaccine communities and the epidemic size. At this stage, also the benchmark campaign is ineffective in mitigating the spread of anti-vaccine propagation. Conversely, when intervention occurs earlier, particularly at $t = 50$ in the left panel of Fig. 3, the optimal campaign (No_Kno.) still performs significantly better. Similar control can be obtained when applying the campaign at $t = 100$, though the epidemic size is relatively larger in this scenario compared to earlier interventions. By examining the clustering of anti-vaccine neighbors in the network in Fig. 2, we see that although clustering begins to emerge at $t = 100$, it is still possible to manage the growth of these clusters with earlier interventions. However, beyond this point, it becomes very challenging to contain or mitigate the negative propagation in the network.

Therefore, effective management of anti-vaccine communities is crucial for minimizing the spread of an epidemic. How can we constrain the growth of these communities to minimize the epidemic spread with late interventions? In the next section, we propose a strategy to mitigate the expansion of unprotected communities by strategically targeting individuals within existing anti-vaccine clusters.

3.2 Cluster-Based Campaign to Constrain Anti-vaccine Community Growth

As demonstrated in previous campaigns, late intervention in applying positive campaigns renders optimal strategies ineffective, even those that work efficiently

when initiated concurrently with the negative campaign. This is attributed to the presence of anti-vaccine clusters, which continue to grow despite campaign efforts to contain them due to budget constraints. Therefore, this section shifts focus from individual-based campaigns to designing a campaign aimed at controlling the growth of these clusters. To begin, we construct a cluster adjacency matrix to represent the candidate targets based on existing anti-vaccine communities. The intuition behind this is to perceive the network as a network of anti-vaccine clusters, with the edges representing the neutral agents connected to these clusters. We refer to these agents as bridge agents or critical agents. For clarification, a schematic illustration of network clusters and how we construct the cluster adjacency matrix m for each candidate node based on the network state is provided in Fig. 4 below. In the cluster adjacency matrix m, columns represent existing anti-vaccine clusters, while rows represent candidate nodes (neutrals) in the network. Consequently, node i adjacent to a cluster j will have the value $m_{ij} = 1$ and $m_{ij} = 0$ otherwise. Additionally, rows indicate whether a node is positioned between multiple anti-vaccine clusters, where $\sum_j m_{ij} > 1$ denotes adjacency to more than one anti-vaccine cluster. If a node i is adjacent to multiple clusters, it functions as a bridge, facilitating the merging of these clusters.



	C1	C2	C3	C4
A	1	0	0	0
B	1	0	0	0
C	0	1	1	0
D	0	0	1	1
E	0	0	1	0
F	0	1	0	0

Fig. 4. Schematic illustration of network clusters and the construction of the cluster adjacency matrix m.

Based on this, our aim is to propose a strategy targeting critical nodes situated between different clusters. To achieve this goal, we need to design an objective function aimed at minimizing the number of bridge (critical) agents, rather than solely minimizing the total number of anti-vaccine adopters. Converting such agents leads to the merging of existing clusters, resulting in the rapid

emergence of large unprotected communities within the network. By targeting these nodes, we can promote vaccine uptake in critical regions, particularly those adjacent to unprotected areas. This campaign, referred to as (Cl.), builds on the proposed campaign (No_Kno.) discussed in the previous section. However, since the negative influence from neighbors within the same cluster is not as significant as the influence from different negative clusters on a specific node i, we will neglect candidate targets that are adjacent to only one anti-vaccine cluster and focus on agents situated between more than one anti-vaccine community. Consequently, let N^0_{Crit} be the set of all agents that are adjacent to more than one anti-vaccine community, i.e., $\{i \in N^0_{Crit} | \sum_j m_{ij} > 1\}$, then the estimated number of critical nodes is $\langle n^-_{crit} \rangle = \sum_{i \in N^0_{Crit}} P^-_{r_i}$.

We demonstrate in Fig. 5, see green bars, that this is an efficient campaign, even with a delayed intervention where many anti-vaccine clusters exist. Initiating barriers between existing anti-vaccine communities, as demonstrated, proves to be an efficient approach in containing the epidemic by preventing the large clustering of unprotected individuals.

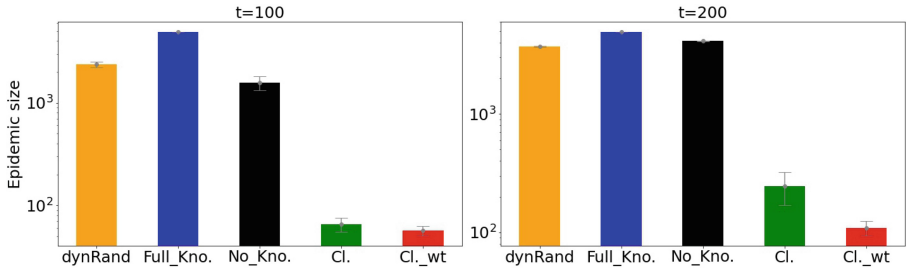


Fig. 5. Comparing optimal cluster-based campaigns to previous campaigns when applying campaigns at time $t = 100$ (left) and $t = 200$ (right). No positive campaign has been applied before t. The results are averaged over 50 simulations, with $\mu^+ = \mu^- = 0.001$, and $\omega = 0.006$.

Moreover, for a more efficient distribution of the limited allocated resources, we must distinguish between different anti-vaccine communities based on their risk when merging with others. Essentially, prioritizing the control of larger communities over smaller ones is crucial to prevent a worsening situation and the emergence of a giant community. Therefore, we also further customize our campaign by introducing weights into the objective function, representing the expected size of the emergent community resulting from the adoption of an anti-vaccine opinion by the bridge node. Through incorporating these weights, we can effectively express the relative importance of the estimated merged cluster sizes that will occur at each bridge node. Thus, the weighted objective function is as follows:

$$\langle n^-_{crit} \rangle_{wgt} = \sum_{i \in N^0_{Crit}} (P^-_{r_i} . \sum_j (|C_j| . m_{ij})), \tag{3}$$

where $|C_j|$ represents the size of cluster j. We can see in Fig. 5, this campaign, referred to as (Cl._wt), leads to further improvement, as depicted by the red bars in the figure, even when interventions are made as late as $t = 200$. By strategically focusing on key areas within the network, the campaign can still achieve substantial results in mitigating the spread of anti-vaccine influence despite the delay.

4 Conclusion

In this study, we explore the potential of mitigating vaccine hesitancy with the ultimate goal of reducing epidemic size. We consider the problem as an optimization problem, aiming to minimize the number of anti-vaccine opinion adopters. We propose an optimal campaign that provides continuous influence allocation to minimize not only the number of anti-vaccine opinion adopters but also the number of bridge agents located in critical regions to mitigate the interconnectivity between anti-vaccine communities. This approach underscores the importance of strategic interventions to combat anti-vaccine diffusion, particularly in scenarios where initial opportunities for early action have been missed. We demonstrate that early interventions can contain negative influence and significantly reduce epidemic size. However, for late interventions, focusing on mitigating anti-vaccine communities to prevent the formation of large clusters is much more efficient for containing infectious diseases.

References

1. WHO. Ten health issues who will tackle this year (2019). https://www.who.int/news-room/spotlight/ten-threats-to-global-health-in-2019
2. Hotez, P., Nuzhath, T., Colwell, B.: COVIRO **41**, 1–7 (2020)
3. Pierri, F., et al.: Sci. Rep. **12**, 5966 (2022)
4. Burki, T.: The lancet digital. Health **1**, e258–e259 (2019)
5. Yin, Q., Wang, Z., Xia, C., Bauch, C.: Commun. Nonlinear Sci. **109**, 106312 (2022)
6. Lee, S.K., Sun, J., Jang, S., Connelly, S.: Sci. Rep. **12**, 13681 (2022)
7. Vinck, P., Pham, P.N., Bindu, K.K., Bedford, J., Nilles, E.J.: Institutional trust and misinformation in the response to the 2018–19 Ebola outbreak in North Kivu, DR Congo: a population-based survey. Lancet Infect. Dis. **19**(5), 529–536 (2019)
8. Campbell, E., Salathé, M.: Sci. Rep. **3**, 1–6 (2013)
9. Romero Moreno, G., Chakraborty, S., Brede, M.: PLoS ONE **16**, e0252515 (2021)
10. Watts, D.J., Strogatz, S.H.: Nature **393**, 440–442 (1998)
11. Dorso, C.O., Medus, A., Balenzuela, P.: Phys. A **482**, 433–443 (2017)
12. Yin, Q., Wang, Z., Xia, C.: Nonlinear Dyn. **111**, 14583–14595 (2023)
13. Yang, L., Li, Z., Giua, A.: In: American Control Conference (ACC) 2019, pp. 5608–5613 (2019)
14. Prieto Curiel, R., González Ramırez, H.: Sci. Rep. **11**, 6626 (2021)
15. Mumtaz, N., Green, C., Duggan, J.: Systems **10**, 50 (2022)
16. Dai, X., Zhu, P., Guo, Y., Wang, Z.: IEEE Access **7**, 61558–61569 (2019)
17. Ancona, C., Iudice, F., Garofalo, F., De Lellis, P.: Sci. Rep. **12**, 11835 (2022)
18. Kermack, W.O., McKendrick, A.G.: Proc. R. Soc. Lond. **115**, 700–721 (1927)

19. Sontag, A., Rogers, T., Yates, C.J.R.: Soc. Interface. **19**, 20210668 (2022)
20. Chen, J., Liu, Y., Yue, J., Duan, X., Tang, M.: Nonlinear Dyn., 1–11 (2022)
21. Centola, D.: Science **329**, 1194–1197 (2010)
22. Romero, D., Meeder, B., Kleinberg, J.: In: Proceedings of the 20th International Conference on World Wide Web, pp. 695–704 (2011)
23. Salathé, M., Khandelwal, S.: PLoS Comput. Biol. **7**, e1002199 (2011)
24. Dubé, E., Vivion, M., MacDonald, N.: Exp. Rev. Vaccines **14**, 99–117 (2015)
25. Salathé, M., Bonhoeffer, S.J.R.: Soc. Interface. **5**, 1505–1508 (2008)
26. Gromis, A., Liu, K.: Pediatrics **149** (2022)
27. Budak, C., Agrawal, D., El Abbadi, A.: Proceedings of the 20th International Conference On World Wide Web, pp. 665–674 (2011)
28. Kabir, K., Kuga, K., Tanimoto, J.: Chaos Solitons Fract. **119**, 180–187 (2019)
29. Huang, H., Chen, Y., Ma, Y.: Appl. Math. Comput. **388**, 125536 (2021)
30. Zhang, H., Zhang, H., Li, X., Thai, M.: In: International Conference on Computational Social Networks, pp. 35–47 (2015)
31. Fügenschuh, M., Fu, F.: Appl. Netw. Sci. **8**, 67 (2023)
32. Alahmadi, S., Hoyle, R., Head, M., Brede, M.: ArXiv Preprint arXiv:2407.02980 (2024)

Combating Toxicity: A Systematic Approach to Model Quarantine Intervention for Varied Toxicity Levels

Emmanuel Addai[1], Nitin Agarwal[1,2](✉), and Niloofar Yousefi[1]

[1] COSMOS Research Center, University of Arkansas, Little Rock, AR, USA
{eaddai1,nxagarwal,nyousefi}@ualr.edu
[2] International Computer Science Institute, University of California, Berkeley, USA

Abstract. This study examines toxicity intensity's impact on its spread using a variable control intervention model. By employing the SEIQR (Susceptible, Exposed, Infected, Quarantine, Recovery) epidemiological modeling, we analyze the propagation of toxicity intensities on social media. Datasets from two content areas on the X platform—the COVID-19 and social movements—are used to validate our model. We conducted the model sensitivity analysis using the Latin hypercube sampling-partial rank correlation coefficient method. The analysis reveals that some parameters positively affect the basic reproduction number $\mathcal{R}_0$, while others negatively affect it. The study finds that dividing data into moderate and high toxicity levels results in lower error rates compared to the model without intensity-based splitting. Based on these findings, specific intervention measures are recommended, offering theoretical support for designing effective online toxicity prevention and control strategies.

Keywords: Toxicity spread · Toxicity intensity · Sensitivity analysis · Reproduction number · Epidemiological model · Quarantine intervention

1 Introduction

Online social media platforms are effective in spreading information about the latest incidents, news, and educational content [1]. However, alongside these positive uses, the dissemination of toxic behavior on social media platforms has become a major concern in today's digital age. With the growing trend of disseminating information through online social media, it is crucial to understand the mechanisms behind the spread of toxic behavior to effectively mitigate its harmful impacts. Given the substantial impact social media platforms have on society, [2] and [3], this understanding can help in developing strategies and tools to detect and counteract toxic behavior, ensuring a healthier and more positive online environment. Consequently, many researchers have developed various methods to understand network structure and user interactions. These methods include data mining, statistical analysis, machine learning algorithms,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 100–112, 2025.
https://doi.org/10.1007/978-3-031-82439-5_9

and social network analysis. By employing these approaches, researchers aim to uncover patterns and dynamics within social media.

Moreover, in recent years, mathematical modeling has become increasingly essential in understanding the dynamics toxic behavior and information flow on social media platforms. Researchers such as [4] -[6] have leveraged several epidemiological models, such as SIR, SIS, and SEIR, to study the propagation of online toxicity. The use of epidemiological modeling arises from the similarities between the dissemination of toxicity on social media and the spread of infectious diseases. Both phenomena involve transmission from one individual to others, and their propagation can be modeled using similar mathematical frameworks.

In this research, we apply and assess the impact of toxicity intensity on its spread across various contexts on a social media platform, using the SEIQR (Susceptible, Exposed, Infected, Quarantine, Recovery) epidemiological model which is a variable control intervention model. We assess the model performance using diverse datasets around COVID-19, protests, and social justice movements. The evaluation includes categorizing the infected individuals into 'high' and 'moderate' toxic groups and aims to discern underlying patterns in toxicity diffusion. This understanding informs providers' decisions on content moderation and user engagement strategies, paving the way to develop more impactful intervention approaches. Additionally, this categorization allows for more efficient resource allocation. High-risk groups can then receive focused attention to manage and reduce toxic content online. Below are the research questions we aim to answer:

RQ1: Do variations in toxicity levels influence its dissemination, and does the degree of toxicity intensity impact the model's performance?

RQ2: What are the key parameters in the SEIQR model that have the most significant impact on the effective reproduction number R_0, and how can these parameters be examined to control the spread of toxicity on social media?

The paper is organized as follows: Sect. 2 reviews related work on toxicity spread in social media, and efforts in applying epidemiological modeling to online platforms. Section 3 outlines the methodology, model formulation, and model analysis. Section 4 explains the findings and the discussion of this research. Finally, Sect. 5 concludes the paper and suggests future directions.

2 Related Work

Recently, the increase in social media toxicity has created significant issues in the digital space, prompting experts to explore its dynamics, effects, and potential solutions. Numerous studies have focused on understanding and classifying toxic behaviors. For example, researchers [7] and [8] sought to compare and assess toxicity levels in COVID-19 discussions across various platforms. Their findings revealed notable differences in the nature and extent of toxicity between these platforms, highlighting the influence of certain key users or groups in spreading toxic discourse and misinformation. Similarly, [9] used machine learning methods to detect various forms of toxic content, including hate speech, harassment, and offensive language. In their analysis of communication patterns on Reddit,

researchers [10] used machine learning to investigate the impact and spread of toxic comments within discussion threads. Additionally, [11] introduced a model to explain how toxicity propagates in a network over time and developed a toxicity contagiousness score.

Epidemiological models for online toxicity compare the spread of harmful content to infectious diseases, suggesting that toxic behavior can infect users, prompting similar actions. Researchers have developed various epidemiological models to predict the future behavior of toxic users and describe their toxicity dynamics. The study by [3] compared different epidemiological models such as SIR (Susceptible-Infected-Recovery), SIS (Susceptible-Infected-Susceptible), and STRS (Susceptible-Toxic-Recovery-Susceptible) for analyzing toxicity and various intensities of toxicity.

Introducing quarantine as a variable control intervention limits the spread of an infectious agent by isolating infected individuals, thereby reducing their contact with those who are susceptible. In epidemiological models, many researchers have adopted this control strategy to study many epidemics, such as [12] where the authors studied the dynamics of Ebola Virus Disease, and [13] where the authors modeled the effects of quarantine rates on COVID-19 in Saudi Arabia. In the context of social network and behavioral analysis, [14] considered fractional-order SEIQR models where quarantine control strategies were utilized. By analyzing the patterns of toxicity spread, insights from epidemiological models offer new strategies for implementing control interventions to mitigate online toxicity.

To the best of our knowledge, no prior studies have investigated the varying levels of toxicity intensity and their impact on toxicity propagation using the SEIQR epidemiological model, a gap this research aims to fill. Furthermore, previous works on SEIQR epidemiological modeling have not addressed sensitivity analysis. This study seeks to identify the key parameters in the SEIQR model that influence the basic reproduction number $\mathcal{R}_0$ and explore how these parameters can be leveraged to control the spread of toxicity on social media.

3 Methodology

This section provides an overview of data collection and the methodology used.

Data. Data from two areas is used to validate our model: the COVID-19 pandemic (spanning from February 2020 to June 2021) and social movements in Brazil (spanning November 1, 2022, to February 25, 2023) and Peru (spanning December 7, 2022, to January 31, 2023). X data is collected using Academic APIs, capturing public discourse on X.

The COVID-19-related data comprises COVID-19, face masks, 5G conspiracy theories, and lockdown. Allegations of electoral fraud in Brazil after the October 30, 2022, presidential elections lead to riots and demands for military intervention by Bolsonaro supporters. In response, pro-government protests emerged, demanding retribution against the rioters. In Peru, anti-government protests

erupted after President Pedro Castillo was removed from power on December 7, 2022, resulting in violent clashes. Some of the popular hashtags used to collect data for each dataset are: #f*ckthecovid/s, #f*ckyourmask/s,#f*cklockdown/s, #5GCoronavirus, #semanistia, #SOSbrasil, #peruprotest/s. In all seven datasets, the Detoxify model calculates the toxicity score for each post. Detoxify uses convolutional neural networks and word vector representations to predict the probability that a given text will be identified as toxic. The model is built on a vast dataset of labeled text, featuring both harmful and benign language. When text is entered into Detoxify, the tool breaks it down into tokens and creates embeddings that represent the semantic meaning and connections between the words. It then provides a toxicity score, ranging from 0 to 1, indicating the likelihood of the text being harmful. We use a threshold of 0.5 for the toxicity score, posts with a score above 0.5 are considered toxic, while those below 0.5 are considered non-toxic [15]. After categorizing data into toxic and non-toxic groups, we further classify the toxic group into high and moderate categories based on intensity. This approach allows us to effectively answer our research questions. A threshold based on the average toxicity score was set across all datasets. Posts below this average were classified as moderately toxic, while those above were deemed high toxic. Table 1 indicates the number of high and moderate toxic posts and the average toxicity. To identify the most active and toxic users, we selected the top 10% of users based on their activity level (e.g., number of retweets) and transferred them to a quarantine compartment. The threshold for defining the most active users is set based on the top 10% of the activity distribution, determined by the number of retweets each post gets [14].

Table 1. Toxicity dataset statistics

Dataset	Num. of Posts	Avg. Toxicity	Num. of High Toxic	Num. of Moderate Toxic
F*Covid	28,131	0.91	4,684	2,082
F*Mask	2,423	0.91	538	217
F*Lockdown	1,995	0.82	598	493
5G	33,403	0.84	1,096	703
Brazil Anti	405,160	0.70	1,221	2,309
Brazil Pro	44,415	0.75	105	131
Peru	195,290	0.71	511	546

Model Formulation. Online toxicity has emerged as a new epidemic, spreading rapidly across digital platforms and affecting individuals and communities worldwide. Its main spread characteristics are high transmission rate, latent nature, and infectiousness of the user in the latent state. Therefore, the SEIQR model is chosen for analysis in this paper. In the traditional SEIQR model, the study population or users are divided into five types: Susceptible (S) represents

susceptible users, that is, users who have never been infected with a toxic post but can be infected by contact with users who post toxic content; Exposed (E) represents latent users, that is, susceptible users who are infected with the toxic posts until they begin to post; Infected (I) represents infected users, that is, users who post toxic content; Quarantined (Q) represents users whose accounts are temporarily suspended by network operators. Recovered (R) represents users who are no longer posting toxic content on social media platforms. Figure 1 indicates the transfer diagram of the SEIQR model. In this model, $\mathcal{A}$ is the recruitment rate of users, that is, new users joining the platform, β represents the effective contact rate, that is, new users joining the platform come in contact with infected users, ψ represents the rate at which the exposed become infected, θ presents the rate at which infected users transfer to quarantine class, η presents recovery rate, γ presents the rate at which quarantine users move to recovery class, ϕ represents the rate at which infected users move to recovery, and μ represents the rate at which people exit autonomously from the platforms. The dynamics of online toxicity spread can be described by the following system of ordinary differential equations.

$$\begin{cases} \frac{dS(t)}{dt} = \mathcal{A} + \eta R - \frac{\beta IS}{N(t)} - \mu S, \\ \frac{dE(t)}{dt} = \frac{\beta IS}{N(t)} - (\mu + \psi)E, \\ \frac{dI(t)}{dt} = \psi E - (\mu + \phi + \theta)I, \\ \frac{dQ(t)}{dt} = \theta I - (\gamma + \mu)Q, \\ \frac{dR(t)}{dt} = \gamma Q + \phi I - (\mu + \eta)R, \end{cases} \tag{1}$$

where $\lambda = \frac{\beta IS}{N(t)}$, the entire population, we define the quantity $N(t)$ by

$$N(t) = S(t) + E(t) + I(t) + Q(t) + R(t).$$

and initial conditions

$$S(0) = S_0 \geq 0, \quad E(0) = E_0 \geq 0, \quad I(0) = I_0 \geq 0,$$
$$Q(0) = Q_0 \geq 0, \quad R(0) = R_0 \geq 0.$$

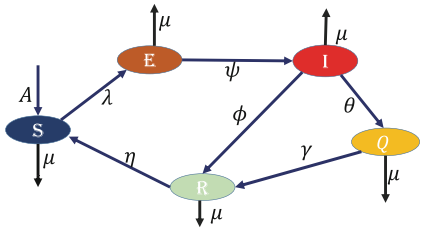


Fig. 1. Transfer diagram for the toxicity spread on the social network platform.

3.1 Model Parameterization and Data Fitting Analysis

Securing robust model validation and achieving optimal parameter estimations are crucial in the field of mathematical modeling based on real data. The key reason for this necessity is the challenge of accurately determining parameter values from the data. A crucial task is to appropriately fit a model's parameters to its specifics. There are certain parameters for models evaluating the dissemination of false information, such as those related to the datasets we have considered.

Given the inherent uncertainties present in data analysis, the potential for inaccuracies in the conclusions cannot be overlooked. While a wide array of methods for estimating parameter values are discussed in academic literature, the least-squares technique is notably the most commonly utilized due to its effectiveness in various contexts, which has the equation:

$$E_{-\mathrm{rel}} = \frac{\|I_{est}(t_i) - I_{data}(t_i)\|_2}{\|I_{data}(t_i)\|_2}. \tag{2}$$

This error rate, $E_{-\mathrm{rel}}$, measures the difference between the estimated number of infected users, $I_{est}(t_i)$, and the actual number of infected users, $I_{data}(t_i)$, at different time points t_i. The norm $\|\cdot\|_2$ represents the Euclidean distance, providing a way to quantify the overall error relative to the actual data. A lower error rate indicates a better fit of the model to the observed data, thereby validating the model's accuracy in representing the spread of toxicity.

3.2 Basic Reproduction Number

The basic reproduction number, represented as $\mathcal{R}_0$, is described as the average number of secondary infections generated by one infected individual within a completely susceptible population. The primary application of $\mathcal{R}_0$ lies in assessing whether a new infectious disease has the potential to proliferate among a population. Furthermore, it helps ascertain the fraction of the population that must be vaccinated to halt the spread of an infectious disease and potentially eliminate it. For our proposed SEIQR model, the system has a unique disease-free equilibrium (E_0) as

$$E_0 = (S, E, I, Q, R) = (\frac{\mathcal{A}}{\mu}, 0, 0, 0, 0)$$

since at disease-free equilibrium $E = I = Q = R = 0$. Now, by applying the next-generation matrix technique to a dynamic system, we derive the

$$\mathcal{R}_0 = \frac{\psi\beta\mathcal{A}}{\mu(\mu+\psi)(\mu+\phi+\theta)}.$$

In widely utilized models of infectious diseases, situations, where the basic reproduction number is greater than 1 ($\mathcal{R}_0 > 1$), suggest that an infection has the potential to proliferate within the population. Conversely, if the basic reproduction number falls below 1 ($\mathcal{R}_0 < 1$), it indicates that an infection is unlikely to initiate a spread among the population.

3.3 Sensitivity Analysis

We begin by identifying the parameter values that are most influential in determining disease dynamics. It is vital to discover numerous aspects that contribute to the virus's transmission and prevalence to decide the best technique for minimizing the number of affected people. Online toxicity spread has been linked to $\mathcal{R}_0$ and those who had been exposed to the toxic contents. The normalized sensitivity index is given by;

$$C_p^{\mathcal{R}_0} = \frac{\partial R_0}{\partial p} \times \frac{p}{\mathcal{R}_0}$$

where p is the parameter being studied for its sensitivity.

3.4 Sensitivity Analysis of Basic Reproduction

The sensitivity index of each model parameter, which is connected with the basic reproduction number, R0, is calculated in this section. This index indicates the relative importance of each parameter in the model that depicts toxicity spread. The index is used to find the parameter with the greatest impact on $\mathcal{R}_0$, which is subsequently utilized as the intervention's target. Parameters with a large impact on $\mathcal{R}_0$ imply that they have a significant impact on online toxicity endemicity. We explore the sensitivity of the parameters by applying the Latin Hypercube Sampling-Partial Rank Correlation Coefficient (LHS-PRCC) method based on the estimated parameter values. The LHS-PRCC are statistical combined methods that are employed in uncertainty analysis to explore the entire parameter space of a model. The LHS assesses parameter values from a multidimensional distribution, and PRCC examines how variations in input parameters correspond with changes in the output of the model. By combining LHS with PRCC to study the sensitivity of the parameters in SEIQR model, we can systematically and comprehensively analyze the the impact of each parameter.

4 Results and Discussion

In order to answer **RQ1**, by employing the outlined methodology, we applied the SEIQR model to seven distinct datasets to assess their performance and alignment with observed data. Table 2 presents the error rates obtained from these evaluations using Eq. 2; it shows the error rate for moderate and high toxicity levels, as well as the error rate without splitting the data. Figures 2a, 2b, and 2c show the error rates and fitted models' representations for the three datasets—5G, Peru, and F*covid—that use the SEIQR model without splitting the data are depicted in. In the figures, the red stars represent the observed data, while the blue line depicts our fitted model. The plot illustrates how well our model aligns with the observed data. The plots in Fig. 3 indicate the fitted plots for the same datasets—5G, Peru, and F*covid—using the SEIQR model with a split in the data based on the toxicity intensity. For simplicity, we excluded

the fitted model for the rest of the datasets and summarized all in Table 2. We compare the error rates of the SEIQR model with and without dataset splitting. The consistently low error rates across datasets when splitting the data into moderate and high toxicity levels, as compared to using the model without this division, underscore the effectiveness of the SEIQR model in capturing the dynamics of toxicity diffusion. The findings suggest that employing the SEIQR model and categorizing toxicity into moderate and high levels is reliable for understanding and predicting the spread of toxicity in various contexts.

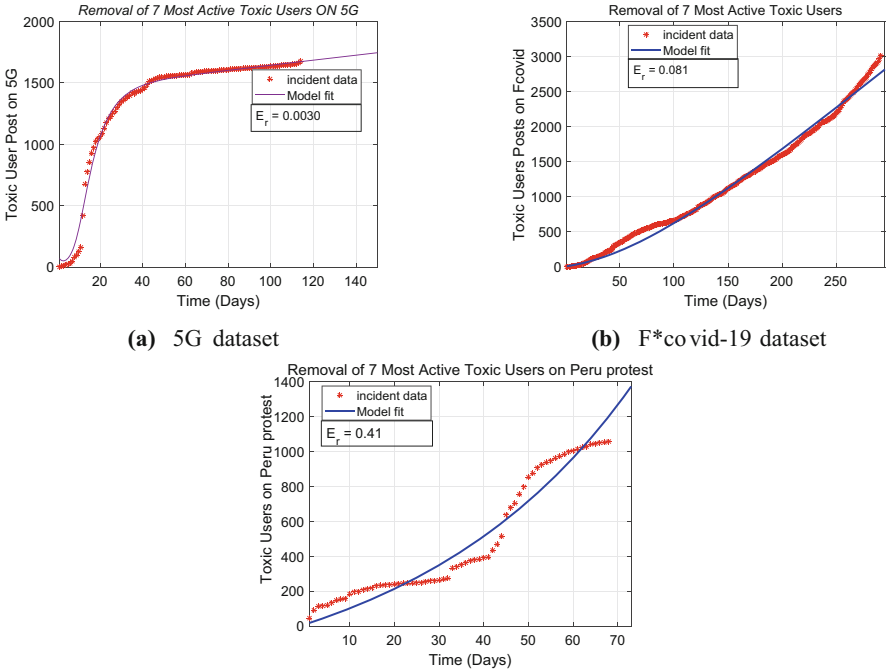
Table 2. Error rates for SEIQR model with and without dataset splitting.

Dataset	Moderate toxicity	High toxicity	Without split
F*Covid	0.0011	0.0021	0.081
F*Mask	0.021	0.049	0.073
Lockdown	0.032	0.045	0.061
5G	0.0029	0.0011	0.003
Brazil Anti	0.062	0.055	0.094
Brazil Pro	0.060	0.061	0.088
Peru	0.095	0.124	0.41

The finding indicates that we can better capture the propagation of toxicity when the dataset is segmented. This pattern is consistent across different contexts, such as COVID-19 and social movements, suggesting that the findings are applicable in various scenarios. This approach provides policymakers with insights on prioritizing areas requiring focused interventions and allocating resources to effectively reduce the spread of toxicity.

4.1 Sensitivity Analysis of Important Parameters

To answer **RQ2**, we begin by identifying the parameter values that are most influential in determining social media toxicity spread. It is vital to discover numerous aspects that contribute to the toxicity spread and prevalence to decide the best technique for minimizing the number of affected users. To determine the dependence of each parameter on the SEIQR model, a sensitivity analysis of each parameter was performed using the Latin Hypercube Sampling-Partial Rank Correlation Coefficient (LHS-PRCC) method. The PRCC corresponds directly to the degree of statistical influence. A positive value indicates that an increase in this parameter leads to a positive influence on the SEIQR model. In contrast, a negative value indicates that an increase in this parameter leads to a negative influence on the SEIQR model. As shown in Fig. 4, among the ϕ, β, ψ, μ, $\mathcal{A}$, and θ parameters, ϕ, β, ψ, $\mathcal{A}$, and θ have positive effects on $\mathcal{R}_0$. In contrast, parameter μ has a negative effect on $\mathcal{R}_0$. The red line represents the p-value, and in the context of social media toxicity $p < 0.05$ indicates that the results of the



(a) 5G dataset **(b)** F*covid-19 dataset

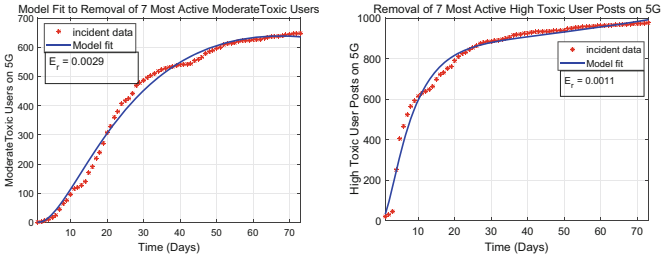
(c) Peru protest dataset

Fig. 2. SEIQR model fitted to infected without splitting toxicity in the datasets.

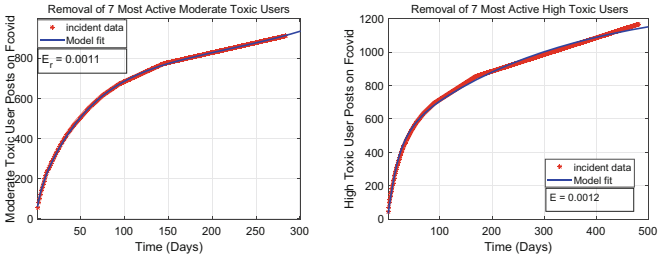
sensitivity analysis are statistically significant. Specifically, it means that there is less than a 5% probability that the observed effects (relationships between the parameters and the effective reproduction number $\mathcal{R}_0$) occurred by chance. These behaviors are significant in executing emergency management measures.

4.2 Emergency Management Measures

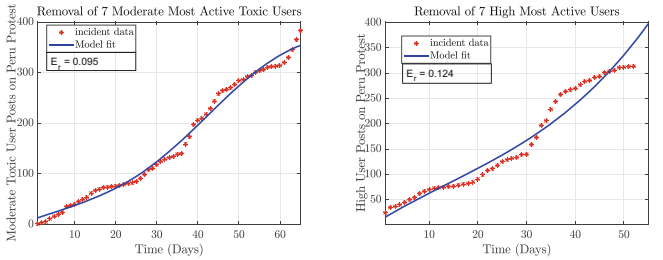
Based on the influence of various parameters on $\mathcal{R}_0$, the following suggestions on social media toxicity contagious emergency management measures are proposed. **First**, implement temporary account suspensions and restricted posting measures. These strategies balance managing harmful behavior with allowing users to reform. This approach prioritizes community safety, promotes positive interactions, and fosters a healthier online environment. **Secondly**, enhancing content demotion and engagement restrictions. Content demotion involves using algorithms to lower the visibility of posts identified as toxic. This ensures such content appears less frequently in users' feeds and is less likely to be seen by a wide audience. Engagement restrictions limit the interaction metrics (e.g., likes, shares, comments) on posts identified as toxic, preventing them from gaining traction and spreading widely. Effective implementation of these strategies to



(a) 5G data (left: moderate toxic, right: high toxic).



(b) F*Co vid data (left: moderate toxic, right: high toxic).



(c) P eru protest data (left: moderate toxic, right: high toxic).

Fig. 3. SEIQR model fitted to moderate and high toxic posts.

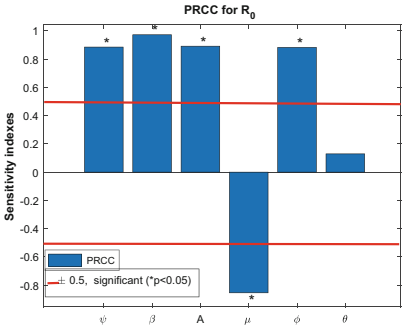


Fig. 4. Sensitivity of $\mathcal{R}_0$ of the online toxicity contagious

mitigate social media toxicity will help limit the spread of harmful content and encourage positive interactions, fostering a safer and more welcoming online community. **Thirdly**, encouraging cross-platform collaboration among network service providers. Toxic content can spread across multiple social media networks, forums, and websites. Collaboration among network service providers ensures a unified approach, resource efficiency, technological innovation, standardized policies and procedures, and comprehensive data sharing, to identify and mitigate harmful behavior, prevent toxic users from simply migrating to another platform to continue their activities. **Lastly**, community quarantined and positive engagement groups, or creating separate discussion groups for users flagged for toxic behavior where they can interact with others under strict moderation. These also direct quarantined users to areas where positive interactions are encouraged and monitored by moderators to promote better behavior.

5 Conclusion and Future Works

In this paper, we utilized the SEIQR epidemiological model to study toxicity on social media platforms. We focused on different intensities of toxicity and applied our model to both high and moderate toxicity levels. The findings show that by categorizing toxicity based on its intensity, we obtained a lower error rate, meaning we were able to capture the propagation of toxicity more accurately. This classification enables providers and policymakers to allocate resources more efficiently, Additionally, high-risk groups can receive targeted attention to manage and mitigate harmful content online. To estimate the model parameters, we fit the model to X data on COVID-19-related and social movements in Brazil and Peru. We carried out a sensitivity analysis to determine the important model parameters that significantly affect the dynamics of toxic spread on social media platforms using the Latin Hypercube Sampling- Partial Rank Correlation Coefficient (LHS-PRCC) method. We observed the most important sensitive parameters. The results show the ϕ, β, ψ, $\mathcal{A}$, are positive sensitivity parameters. Conversely, parameter μ is a negative sensitivity parameter. In the context of toxicity spread on social media, understanding these sensitivities helps identify crucial factors that either increase or decrease the spread of harmful content. This knowledge is essential for developing effective strategies to reduce online toxicity. Based on the impact of sensitive parameters on basic reproduction number ($\mathcal{R}_0$), this paper makes recommendations in terms of strengthening the implementation of temporary account suspension and restricted content, enhancing content engagement restrictions, encouraging cross-platform collaboration, and creating separate discussion groups for users flagged for toxic behaviors. These provide a reference value for developing reasonable and practical emergency management measures for online toxicity. Future studies could analyze the real-time monitoring systems based on the SEIQR model to identify early signs of toxicity spread and initiate immediate intervention measures. We aim to apply the SEIQR model to different social media platforms to evaluate if the model parameters and sensitivity analysis results hold across diverse online environments.

Acknowledgments. This research is funded in part by the U.S. National Science Foundation (OIA-1946391, OIA-1920920), U.S. Office of the Under Secretary of Defense for Research and Engineering (FA9550-22-1-0332), U.S. Army Research Office (W911NF-23-1-0011, W911NF-24-1-0078), U.S. Office of Naval Research (N00014-21-1-2121, N00014-21-1-2765, N00014-22-1-2318), U.S. Air Force Research Laboratory, U.S. Defense Advanced Research Projects Agency, Arkansas Research Alliance, the Jerry L. Maulden/Entergy Endowment at the University of Arkansas at Little Rock, and the Australian Department of Defense Strategic Policy Grants Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations. The researchers gratefully acknowledge the support.

References

1. Pew Research Center: Experiencing online harassment (2020)
2. Benevenuto, F., Rodrihues, T., Cha, M., Almeida, V.: Characterizing user behavior in online social networks. In: Proceedings of ACM SIGCOMM International Measurement Conference (2009)
3. Shajari, S., Alassad, M., Agarwal, N.: Characterizing suspicious commenter behaviors. In: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, pp. 631–635. ACM, Kusadasi Turkiye (2023)
4. Dai, L., Liu, X., Chen, Y.: Global dynamics of a fractional-order sis epidemic model with media coverage. Nonlinear Dyn. **111**, 19513–19526 (2023)
5. DiCicco, K., Noor, N. B., Yousefi, N., M., Spann, B., Agarwal, N.: Toxicity and networks of covid-19 discourse communities: a tale of two social media platforms. In Proceedings of the 3rd Workshop on Reducing Online Misinformation through Credible Information Retrieval 2023, pp. 30–42 (2023). ISSN 1613, 0073
6. Yousefi, N., Agarwal, N.: Studying the influence of toxicity intensity on its propagation using epidemiological models. In: AMCIS 2024 Proceedings (2024)
7. DeMarsico, D., Bounoua, N., Miglin, R., Sadeh, N.: Aggression in the digital era: assessing the validity of the cyber motivations for aggression and deviance scale. Assessment **29**(4), 764–781 (2022)
8. Noor, N.B., Yousefi, N., Spann, B., Agarwal, N.: Comparing toxicity across social media platforms for covid-19 discourse. In: The Proceeding of the Ninth International Conference on Human and Social Analytics, pp. 21–26 (2023)
9. Yousefi, N., Noor, N.B., Spann, B., Agarwal, N.: Towards developing a measure to assess contagiousness of toxic tweets. In: Proceedings of the International Workshop on Combating Health Misinformation for Social Wellbeing (2023)
10. Yousefi, N., Noor, N.B., Spann, B., Agarwal, N.: Examining toxicity's impact on Reddit conversations. In: Cherifi, H., Rocha, L.M., Cherifi, C., Donduran, M. (eds.) Proceedings of the Complex Networks & Their Applications XII, pp. 401–411. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-53503-1_33
11. Li, T., Wang, S., Li, B.: Research on suppression strategy of social network information based on effective isolation. Procedia Comput. Sci. **131**, 131–138 (2018)
12. Ngwa, G.A., Teboh-Ewungkem, M.I.: A mathematical model with quarantine states for the dynamics of ebola virus disease in human populations. Comput. Math. Methods Med. (2016)

13. Alsheri, A.S., Alraeza, A.A., Afia, M.R.: Mathematical modeling of the effect of quarantine rate on controlling the infection of covid-19 in the population of saudi Arabia. Alex. Eng. J. **61**(9), 6843–6850 (2022)
14. Addai, E., Yousefi, N., Agarwal, N.: SEIQR: an epidemiological model to contain the spread of toxicity using memory-index. In: Proceedings of 18th International AAAI Conference on Web Social Media: CySoc, 5th International Workshop Cyber Social Threats 2024, pp. 1–12 (2024)
15. Hanu, L.: Unitary, team: Detoxify (2020). https://github.com/unitaryai/detoxify,

Modeling Toxicity Propagation on Reddit Using Epidemiology

Selimhan Dagtas[1], Nitin Agarwal[1,2(✉)], and Niloofar Yousefi[1]

[1] COSMOS Research Center, University of Arkansas – Little Rock, Little Rock, USA
{sedagtas,nxagarwal,nyousefi}@ualr.edu
[2] International Computer Science Institute, University of California, Berkeley, USA

Abstract. Understanding the propagation of toxicity on social media platforms is necessary for ensuring safer digital spaces. This paper aims to measure the accuracy of the five most established epidemiological models on the spread of toxicity in Reddit. By applying these models to two datasets—one from a highly toxic subreddit that was banned and one from various groups throughout Reddit—we address a gap in the literature by evaluating their performance on Reddit data. The results indicate that epidemiological modeling can achieve a fitting error of less than 2% on toxicity spread in Reddit. Furthermore, we concluded that more sophisticated models such as SEIRS and SIRS performed significantly better than simpler models like SIS or SIR. This research successfully demonstrates the effectiveness of epidemiological modeling on Reddit and paves the way for further research and potential prevention strategies for toxicity on Reddit.

Keywords: Epidemiological · Toxicity · Reddit · Modeling · SIS · SIR · SIRS · SEIR · SEIRS

1 Introduction

Toxicity on social media is becoming increasingly viral and, as users spread toxicity, it has become a propagating problem. The first step in mitigating toxicity on social media is identifying where it is and understanding how it spreads. Various studies have analyzed toxicity across platforms and have concluded that "Reddit is the most toxic social media platform" [9]. Hence, in this research, we focus on studying the spread of toxicity on Reddit. Now that a platform that propagates toxicity is identified, the next step is to find a method to understand how it spread. Epidemiological modeling has been and is still being used to study the spread of viral diseases throughout the world [4]. Epidemiological models have been proven to be effective at modeling and predicting the spread of harmful behavior online [10]. Toxicity in social media has been previously modeled using epidemiological modeling on different platforms. Yet, these methods have never been used to model and study the propagation of toxicity on Reddit, which is notorious for having some of the most toxic posts and comments. If these models

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 113–124, 2025.
https://doi.org/10.1007/978-3-031-82439-5_10

can accurately model and help us understand toxicity propagation on Reddit we can combat toxicity more effectively. This was the primary motivation for this study and the main driver behind the following research questions:

RQ1: Can toxicity propagation on Reddit be fitted accurately by applying epidemiological models?

RQ2: Are more complex epidemiological methods better at modeling toxicity propagation on Reddit?

2 Literature Review

Epidemiological models were initially used for understanding the infectiousness of medical diseases. Especially viral diseases with high rates of infection have been analyzed and understood better using epidemiological modeling. Recently, the SEIR model was used by researchers to model the infectiousness of COVID-19 and mathematically forecast COVID-19 epidemic situation [8].

Epidemiological models for modeling toxicity on social media have predominantly been used on X, formerly known as Twitter. One study in particular inspired our research: a study that applied three different epidemiological models to toxic Twitter posts and comments [16]. Their paper used three of the core epidemiological models: SIS, SIR, and SIRS. These models are the foundational models that other epidemiological models have been built on. Furthermore, a model developed by Yousefi et al. builds upon the well-known previous SEIR and SEIRS models by separating the infected into two separate categories [18].

There are many epidemiological models that have been developed, but, to ensure the most consistent results, it is preferable to stick to the most popular models. Furthermore, these models have been applied to a verity of data whether it is users, posts, topics or hashtags. For example, Jonathan Skaza successfully used the SIR to model the proliferation of Twitter hashtags and gain insights on their infectiousness [15].

Even though epidemiological modeling has not been applied to Reddit, toxicity on Reddit has been studied. Specifically, the tree structure of its posts and comments have been analyzed to study the spread of toxicity and predict its occurrence [6,17]. That research contained useful insights on how to find and collect the most toxic data on Reddit and also how to reprocess it for analysis.

This study fills a critical gap in the literature by applying and evaluating established epidemiological models to the spread of toxicity on Reddit for the first time. While prior research has applied these models to other social media platforms, there has been a lack of focus on Reddit, which is a platform with unique community dynamics and that is notorious for being one of the most toxic social media platforms.

3 Methodology

In this section, we explain in detail the methodology that was used to collect data from Reddit and to study toxicity propagation using epidemiological modeling.

3.1 Data Collection

The first step in the data collection process was to identify the data to be collected. Since the study focuses on toxicity propagation, we prioritized datasets with a significant presence of toxic content. Anti-COVID groups on social media have been notorious for being toxic, as stated by Pascual-Ferrá [11], and were included and targeted in one dataset. Additionally, we selected a publicly available dataset related to climate change to ensure a broader analysis. To address potential concerns about the credibility of the study, we used publicly available datasets. Hence, this study utilized two datasets, both sourced from different subreddits on Reddit. Since these datasets cover different types of content, they contributed to improving the generalizability of our model.

The first dataset selected and collected was from the subreddit "r/NoNewNormal". This subreddit was involved in spreading misinformation about COVID-19, violating the platform's policies on health-related disinformation. This community was infamous for opposing public health measures, such as mask mandates, lock-downs, and vaccinations. This subreddit was banned and removed from the platform shortly after it gained popularity. The data was collected using the Pushshift API, which collected all of the posts and comments from the subreddit. The API was accessed and executed using Python's PSAW library, following the steps in the documentation. This dataset had a total of 2.2 million posts and comments dating from June 2020 to August 2021.

The second dataset was selected from one of the most common data hubs used by data scientists, Kaggle [1]. Among the toxic datasets available on Kaggle, the "Public Opinion on Climate Change" dataset was chosen for two reasons. Firstly, it contained a moderate balance of both having heated discourse and being relatively representative of an average subreddit on the platform. Choosing a heavily moderated and non-toxic subreddit would not have given us enough data to work with, and an extremist subreddit would not have been a precise representation of toxicity propagation on the platform in general. Secondly, it was an up-to-date dataset that was being regularly updated, resulting in data that represented the most recent toxicity status on Reddit. This dataset had more than 700 thousand entries dating from July 2023 to September 2024. Note that the "Public Opinion on Climate Change" dataset covers a broader date range than what was used in this study. However, because daily toxicity posts prior to a certain date were predominantly 1 and 0, we began our modeling from the first day on which daily toxicity surpassed single digits.

3.2 Data Processing

The datasets collected using both the Pushshift API and Kaggle had faulty or missing information. Therefore, both of these datasets were cleaned up by using filterers that removed null values, deleted comments, and removed posts. These filters guaranteed the toxicity scores calculated in the following steps were from valid posts that existed on Reddit.

This filtered data was then put through a toxicity model to differentiate between toxic and non-toxic posts. The model Detoxify was used to get the toxicity scores, due to its extensive track record and its consistent results since its release date in 2020 [7]. The Detoxify model outputs a variety of toxicity categories, including obscene, severe toxic, threat, insult, and identity hate. It also provides an overall toxicity score that was used in this research. Detoxify provides a score ranging from 0 to 1, where a score closer to 1 indicates a higher likelihood of the post being toxic, and a score closer to 0 suggests a greater likelihood of the post being non-toxic. Based on prior research by authors in [12], a threshold of 0.5 was selected to differentiate between toxic and non-toxic posts. Using that threshold as a toxicity filter left us with 219,577 toxic entries in the No New Normal dataset and 49,208 for Climate Change.

3.3 Epidemiological Models

Epidemiological models were initially used for modeling the spread of viral diseases throughout society. Similarly, researchers treat a toxic post as analogous to a disease and can use epidemiological models to study the propagation of toxicity on social media. These models use various categories paired with a system of differential equations that determine the change of populations over time in each of these categories. Each model is named using both the names of the categories and the direction of flow in the population. In this study, we use established models used for modeling the spread of toxicity in social media. These models were historically formulated in sequential order from the simplest, SIS, to the most complex, SEIRS, due to the increased accuracy that comes with a more complex model. Below are the models from increasing complexity in the same order that they were formulated in.

SIS: The SIS model is one of the most basic epidemiological models [14]. It consists of two groups: susceptible (S) and infected (I). The infected group refers to individuals who, depending on the context, have been infected by a disease or infectious condition. In this case, they post toxic content and are responsible for propagating toxicity. Each toxic comment or post is considered to contribute to the propagation of toxicity. The susceptible are vulnerable to infection and, as time passes, they will become infected. However, occasionally an infected individual will rid themselves of the infection and become susceptible again. This dichotomy is typically modeled using the two parameters beta (β) and gamma (γ). β represents the rate at which the susceptible get infected, and γ represents the rate at which the infected become susceptible again. Nevertheless, recent epidemiological modeling methods have introduced a correction to this representation. The rate at which the susceptible became infected was represented by $\beta * S$, but in reality, the number of infected should have a notable impact on the rate at which the susceptible are infected. Hence, the typical parameter β has been substituted with θ which is calculated by $\beta * \frac{I}{N}$ as shown in Fig. 1. Here, N is the total population, meaning the sum of all the categories in the model and in this case $S + I$, thus the percentage of the total population that has been infected plays an important role in modeling the rate at which the susceptible are

infected. This change is reflected in the system of ordinary differential equations as seen in system (1).

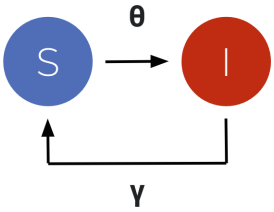


Fig. 1. SIS Model

$$\begin{aligned} \frac{dS}{dt} &= \gamma I - \beta \frac{I}{N} S, \\ \frac{dI}{dt} &= \beta \frac{I}{N} S - \gamma I. \end{aligned} \tag{1}$$

SIR: The SIR model builds on top of the SIS model [5] by introducing a new category: recovered (R). The logic behind this modification is the fact that, if an individual has recovered from being infected, then they will be immune to becoming infected again due to their previous experience. This modification breaks the loop and creates a one directional flow, as seen in Fig. 2. This can be modeled by adding an additional ordinary differential equation to the system as seen in system (2).

Fig. 2. SIR Model

$$\begin{aligned} \frac{dS}{dt} &= -\beta \frac{I}{N} S, \\ \frac{dI}{dt} &= \beta \frac{I}{N} S - \gamma I, \\ \frac{dR}{dt} &= \gamma I. \end{aligned} \tag{2}$$

SIRS: The SIRS model builds further on the SIR model [3]. It refutes the assumption that once recovered, there is no possibility of being infected again.

The SIRS model makes an argument for the probability that, after some time, the recovered could transition back to susceptible. Introducing this parameter allows the model to fit a wider range of data, whether they are strongly curved or linear. This loop back is represented by the parameter delta (δ), portrayed in Fig. 3. The flow from recovered to susceptible is calculated as $\delta * R$ as seen in (3).

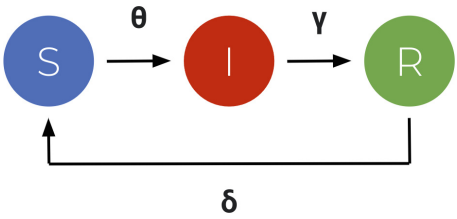


Fig. 3. SIRS Model

$$\begin{aligned} \frac{dS}{dt} &= \delta R - \beta \frac{I}{N} S, \\ \frac{dI}{dt} &= \beta \frac{I}{N} S - \gamma I, \\ \frac{dR}{dt} &= \gamma I - \delta R. \end{aligned} \tag{3}$$

SEIR: The SEIR model, unlike the previous models, introduces two modifications to the previous epidemiological model [8]. Building on top of the SIR model, the SEIR model introduces a new epidemiological category: exposed (E). The introduction of this new category understands that not everyone exposed to infectious content will immediately become infected. Therefore, the exposed category is a buffer layer between the susceptible and the infected. This buffer allows the model to fit data with a more gradual propagation of toxicity. The flow from exposed to infected is modeled as $\sigma * E$. Furthermore, the SEIR model introduces the idea that, for a given platform, there will be users entering and leaving the platform. This idea is modeled using two new parameters Lambda (Λ) and mu μ. Λ is used to model the entering users, which are assumed to be initially susceptible, and μ is used to model the exiting users, which are assumed to exit from each category with an equally proportionate rate as visualized in Fig. 4. This model can be represented mathematically with the system of ODEs seen in system (4).

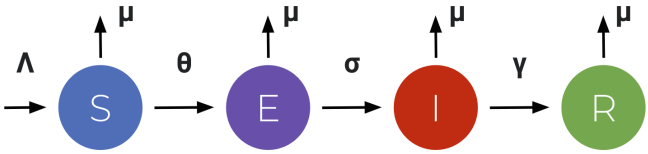


Fig. 4. SEIR Model

$$
\begin{aligned}
\frac{dS}{dt} &= \Lambda - \frac{\beta I}{N} S - \mu S, \\
\frac{dE}{dt} &= \frac{\beta I}{N} S - (\mu + \sigma) E, \\
\frac{dI}{dt} &= \sigma E - (\mu + \gamma) I, \\
\frac{dR}{dt} &= \gamma I - \mu R.
\end{aligned} \tag{4}
$$

SEIRS: The SEIRS model builds upon the SEIR model in a similar fashion as to how the SIRS model built on top of the SIR model [2]. The SEIR model is not a loop and has a one directional and fundamentally limited flow. The SEIRS model introduces the same type of loop back from the recovered to the susceptible category as in the SIRS model. This is modeled using the same parameter as before, δ. This makes the model more accurate at fitting more complex toxicity propagation datasets; however, an astronomical improvement on a relatively linear toxicity spread should not be expected. This loop back seen in Fig. 5 can be added to the system of differential equations as demonstrated in system (5).

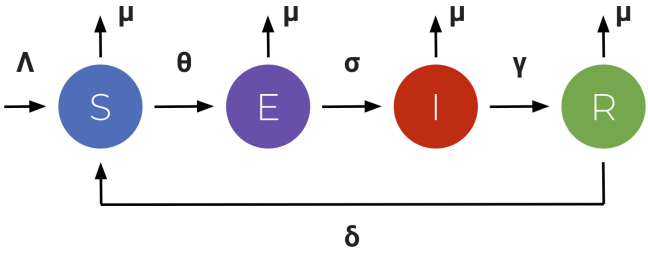


Fig. 5. SEIRS Model

$$\begin{aligned}
\frac{dS}{dt} &= \Lambda + \delta R - \frac{\beta I}{N} S - \mu S, \\
\frac{dE}{dt} &= \frac{\beta I}{N} S - (\mu + \sigma) E, \\
\frac{dI}{dt} &= \sigma E - (\mu + \gamma) I, \\
\frac{dR}{dt} &= \gamma I - (\mu + \delta) R.
\end{aligned} \tag{5}$$

3.4 Parameter Optimization

Now that the models are summarized and formulated into systems of ordinary differential equations, we fit these models to the two datasets previously collected. This model fitting process is called parameter optimization, since the goal of this process is optimize the parameters of each model to fit the data with as low of an error as possible.

Parameter optimization is a well-known and diverse process, meaning that it is used in many different fields and can be done in a variety of ways. One of the most optimal methods of optimizing the models' parameters would be to use a mathematical curve fitting algorithm such as least-squares regression. However, algorithmic methods as such only work on a linear system of equations. In our case, we have an integration of a system of differential equations. Therefore, a more computational approach must be taken.

The typical approach, and the approach that was taken in this research, is to define an error function and apply error minimization algorithms to find the parameters that result in the lowest error when compared with the collected data. We defined the error in a simple equation as shown in Eq. (6).

$$Error = \frac{|Model - Data|}{|Data|} \cdot 100\% \tag{6}$$

One thing to note about parameter optimization is that the complexity of the optimization process increases exponentially with the number of parameters. Therefore, there arises a need to adjust the parameter optimization methods according to the number of parameters in the model being optimized.

For the parameter optimization process we decided to use the Python library Scipy. This library has a function called minimize. Scipy's minimize does exactly as one would expect: it finds the correct variables to minimize a given function. This was chosen specifically due to the included functionality of changing the optimization algorithm. All of the algorithms available are specified in the documentation on the Scipy documentation [13], but for this study, after experimentation, we decided to use the "Nelder-Mead" and the "Limited Memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS-B)" algorithms due to their accuracy. Nelder-Mead's algorithm was used to to calculate the models with three or fewer parameters: SIS, SIR, and SIRS. The L-BFGS-B algorithm was used to

calculate an initial point that was further optimized by Nelder-Mead's algorithm for the more complex models.

4 Results

Before analyzing the results, the expected outcomes were that the models would drop the error rate to at least 10 percent and that the models with higher complexity would fit the data better and have a lower error. This, of course, is the general assumption when it comes to epidemiological models. However, the comparison of the results between the models is highly dependent on the curvature of the toxicity propagation within the dataset.

Table 1. Epidemiological Modeling Errors

Model	No New Normal	Climate Change
SIS	18.51%	24.64%
SIR	7.76%	6.92%
SIRS	5.68%	1.89%
SEIR	4.15%	2.87%
SEIRS	4.13%	2.86%

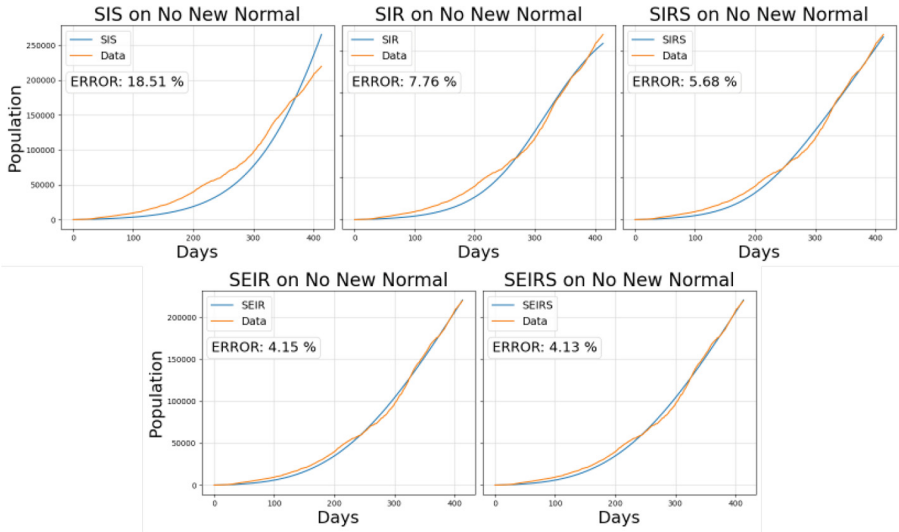


Fig. 6. Epidemiological Modeling on No New Normal dataset

Taking a look at Table 1, there is an obvious trend of decreasing error values as the complexity of the models increases. Furthermore, the error values that the models were able to achieve were pleasantly surprising, thereby answering RQ1, i.e., toxicity propagation on Reddit can be fitted accurately by applying epidemiological models. However, the Climate Change modeling resulted in an increase in the error when upgrading from the SIRS model to the SEIR model. The reason for this could be that when the toxicity propagation is almost perfectly linear, simpler models tend to fit the data better than complex models.

Moving on to the plots of the models, The orange line labeled "Data" is the cumulative toxicity measured in the respective subreddit while the blue line labeled with the corresponding model used represents the output from that model. In Fig. 6 it is once again evident that the more complex models are able to analyze the spread of toxicity much better than the rudimentary models. Furthermore, one can quite clearly see how effective epidemiological models can be in representing the spread of toxicity effectively.

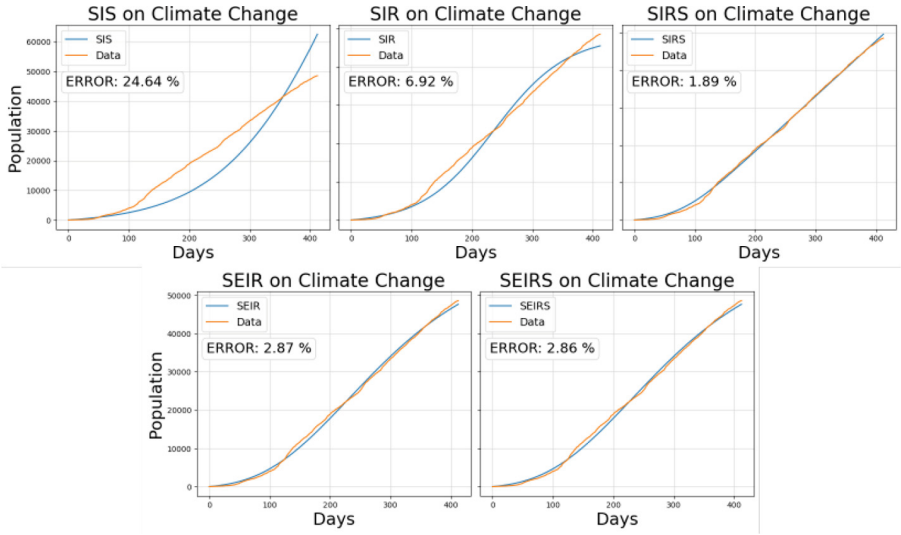


Fig. 7. Epidemiological Modeling on Climate Change dataset

When analyzing Fig. 7, it is again clear that the more sophisticated models generally do a better job at modeling the toxicity propagation in Reddit, thereby answering RQ2. What can be seen here that is not so evident in Table 1 is the linearity of the dataset. Especially when looking at the SIRS line plot, it becomes clear that the data, after the first 100 d, becomes remarkably linear. Thus, simpler models such as SIRS perform well when modeling such linear datasets. These results emphasize the need for examining the data characteristics before applying complex modeling methods.

5 Conclusion

In this research paper, we have successfully shown that modeling the spread of toxicity, using epidemiology, is possible on Reddit. Not only did we show it was possible but we analyzed in detail which of the most popular five epidemiological models performed best and which models should be used in which circumstances. This was done by first collecting large amounts of data from two unique sources for two distinct aims. Then, we obtained the toxicity propagation using the Detoxify library and prepared it for epidemiological modeling. Finally, we tested the top five renowned epidemiological models on both of these datasets to achieve our results.

To answer the initial research questions, and to draw conclusions from this research, it is now clear that epidemiological modeling on Reddit is possible and that increasing the complexity of the models generally increases the accuracy of the fitting. There is one caveat to this conclusion, which is the fact that if the dataset is primarily linear, a more simple model may perform marginally better. Nevertheless, the more complex ones, in particular SEIRS, are more reliable at modeling toxicity propagation.

This research makes significant contributions to the study of toxicity propagation on social media by applying and evaluating five epidemiological models on Reddit datasets. It advances the understanding of how well popular epidemiological models perform on toxicity spread on Reddit. By comparing the error rates of various approaches, this work provides valuable guidance for selecting appropriate models for similar studies, enhancing the robustness of future research. Moreover, the insights gained from this study can inform policymakers and social media platforms in developing more effective intervention strategies and content moderation policies to combat toxic behavior.

Using this research, toxicity in Reddit could be combated or suppressed before it gets out of hand. This leads us to possible future work; now that we can accurately model toxicity propagation on Reddit, what can be done to combat it and make Reddit a more prosperous platform? Future work could include solutions to this question, such as utilizing quarantine measures for proactive policymaking.

Acknowledgements. This research is funded in part by the U.S. NSF (OIA-1946391, OIA-1920920), OUSD (FA9550-22–1-0332), ARO (W911NF-23–1-0011, W911NF-24–1-0078), ONR (N00014-21–1-2121, N00014-21–1-2765, N00014-22–1-2318), AFRL, DARPA, Arkansas Research Alliance, the Jerry L. Maulden/Entergy Endowment at UALR, and the Australian Department of Defense Strategic Policy Grants Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations. The researchers gratefully acknowledge the support.

References

1. Asaniczka: Public opinion on climate change (2023). https://www.kaggle.com/datasets/asaniczka/public-opinion-on-climate-change-updated-daily
2. Bjørnstad, O.N., Shea, K., Krzywinski, M., Altman, N.: The SEIRs model for infectious disease dynamics. Nat. Methods **17**(6), 557–559 (2020)
3. Chen, Y., Yang, J., Zhang, F.: The global stability of an sirs model with infection age. Math. Biosci. Eng. **11**(3), 449–469 (2013)
4. Ciannella, S., González-Fernández, C., Gomez-Pastora, J.: Recent progress on wastewater-based epidemiology for COVID-19 surveillance: a systematic review of analytical procedures and epidemiological modeling. Sci. Total Environ. **878**, 162953 (2023)
5. Cooper, I., Mondal, A., Antonopoulos, C.G.: A SIR model assumption for the spread of COVID-19 in different communities. Chaos Solitons Fractals **139**, 110057 (2020). https://doi.org/10.1016/j.chaos.2020.110057
6. Falade, T.C.C., Yousefi, N., Agarwal, N.: Toxicity prediction in reddit. In: AMCIS 2024 Proceedings, 17 (2024)
7. Hanu, L., Team, U.: Detoxify: pretrained transformer models for toxicity detection (2024). https://github.com/unitaryai/detoxify
8. He, S., Peng, Y., Sun, K.: SEIR modeling of the COVID-19 and its dynamics. Nonlinear Dyn. **101**, 1667–1680 (2020)
9. Noor, N.B., Yousefi, N., Spann, B., Agarwal, N.: Comparing toxicity across social media platforms for COVID-19 discourse. In: Ninth International Conference on Human and Social Analytics (HUSO 2023), pp. 21–26 (2023)
10. Ojha, R.P., et al.: Controlling of fake information dissemination in online social networks: an epidemiological approach. IEEE Access **11**, 32229–32240 (2023)
11. Pascual-Ferrá, P., Alperstein, N., Barnett, D.J., Rimal, R.N.: Toxicity and verbal aggression on social media: polarized discourse on wearing face masks during the COVID-19 pandemic. Big Data Soc. **8**(1), 20539517211023532 (2021)
12. Saveski, M., Roy, B., Roy, D.: The structure of toxic conversations on Twitter. In: Proceedings of the Web Conference 2021, pp. 1086–1097 (2021)
13. SciPy Developers: SciPy minimize function (2024). https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html
14. Shi, H., Duan, Z., Chen, G.: An SIS model with infective medium on complex networks. Phys. A **387**(8), 2133–2144 (2008). https://doi.org/10.1016/j.physa.2007.11.048
15. Skaza, J., Blais, B.: Modeling the infectiousness of Twitter hashtags. Phys. A **465**, 289–296 (2017). https://doi.org/10.1016/j.physa.2016.08.038
16. Yousefi, N., Agarwal, N.: Study the influence of toxicity intensity on its propagation using epidemiological models. In: AMCIS 2024 Proceedings, 17 (2024)
17. Yousefi, N., Noor, N.B., Spann, B., Agarwal, N.: Examining toxicity's impact on reddit conversations. In: International Conference on Complex Networks and Their Applications, pp. 401–411. Springer (2023)
18. Yousefi, N., Agarwal, N., Addai, E.: Developing epidemiological models with differentiated infected intensity. In: 17th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation (2024)

Dynamics on/of Networks

Classification of Sleep Stages Based on Quantile Graphs from Mouse EEG

Kazuki Koyama[1](✉), Masanori Sakaguchi[2], and Takaaki Ohnishi[1]

[1] Graduate School of Artificial Intelligence and Science, Rikkyo University, Tokyo, Japan
23wr005j@rikkyo.ac.jp

[2] International Institute for Integrative Sleep Medicine (WPI-IIIS), University of Tsukuba, Ibaraki, Japan

Abstract. We conduct an analysis to classify sleep stages using electroencephalogram (EEG) data from mice, focusing on Quantile Graphs. The method constructs networks from EEG time series data and uses the extracted network features to classify sleep stages through a random forest. A key aspect of the proposed approach is the inclusion of network-based features such as clustering coefficients and average shortest path length, in addition to traditional statistical features. Since the Quantile Graph is derived from network data, it produces distinct results that differ from those obtained using traditional features. The experiments demonstrate a notable improvement in classification accuracy for rapid eye movement (REM), while the network features are less effective for other stages, such as non-REM (NREM), high frequency theta wave (HT), and low frequency theta wave (LT). The results indicate that the choice of time delay and quantization settings plays a crucial role in achieving accurate classification of sleep stages, with the optimal time delay for maximum classification performance for REM identified at approximately 83.3 ms.

Keywords: electroencephalography · complex network · Quantile Graph · sleep stage classification

1 Introduction

Many researchers engage with a wide range of data from various disciplines. A complex system is defined as a system comprising a number of elements that interact with each other in a multitude of ways. Complex systems have been the subject of research in a diverse range of fields, from biology [7,9], physics [11,15], and economics [8]. A variety of analytical approaches have been utilized to examine these systems, including the construction of dynamic mathematical models. Notable examples are the Hodgkin-Huxley model in biology and stochastic models in other domains. Complex networks have been applied to various time series data and the methods to construct networks from time series has

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 127–138, 2025.
https://doi.org/10.1007/978-3-031-82439-5_11

been proposed [18]. The network features such as degrees, average shortest path length, global clustering coefficient, number of communities, and modularity are widely used in these applications. The popular methods to construct network from time series are Natural Visibility Graph, Horizontal Visibility Graph and Quantile Graph. Natural Visibility Graph is used for a diagnosis of Alzheimer's disease from EEG [1] and the classification of sleep stages from a single channel time series EEG [20]. Horizontal Visibility Graph is used for the classification of seizure, during seizure and healthy from EEG [14]. Natural Visibility Graph and Horizontal Visibility Graph are both utilized for the classification of sleep stages from human EEG data [21]. Quantile Graph is employed for the diagnosis of Alzheimer's disease [16] and epilepsy from EEG [3]. An approach called NetF [19] is proposed, which incorporates Natural Visibility Graph, Horizontal Visibility Graph, and Quantile Graph methods. This approach generates features from networks constructed using these graph techniques.

A Quantile Graph is constructed by partitioning the data set into q discrete values, thereby enabling the visualization of the transition of each value [4,17]. The network is constituted by both a direction and a magnitude of time series. The network features exhibit distinctive characteristics when compared to conventional features derived from time series, such as entropy or statistics. This is due to the inherent differences in the analytical methods used. Various analytical approaches have been applied to electroencephalography (EEG) [2,6,12]. The results of the analysis of EEG data, which is primarily studied by a deterministic chaotic approach, are different when approached by a network based method. In this paper, we classify sleep stages by a random forest using network features of a Quantile Graph.

2 Data and Method

2.1 Data

The data set is EEG recordings from mice. They are sampled at a rate of 512 Hz over a period of approximately 6 hours. The recordings are labeled by skilled workers as sleep and arousal in 5402 epochs. We conduct analysis at each epoch which is a 4-second interval. Sleep and arousal are classified into rapid eye movement (REM) and non-REM (NREM) stages, and further characterized by high frequency theta waves (HT) and low frequency theta waves (LT), respectively. HT and LT are differentiated by the strength of theta wave oscillations, which predominantly observed in the 4 to 10 Hz range. Figure 1 illustrates the typical EEG time series for each sleep stage. The dataset contain 457, 3197, 410, and 1338 epochs classified as REM, NREM, HT, and LT, respectively. The amplitude and frequency of EEG signals vary depending on the sleep stage, with the amplitude being highest during NREM sleep. The waves consist of a wide range of frequencies. Theta waves are most commonly observed during both REM sleep and arousal, particularly in the HT region. Therefore, categorizing the 4 sleep stages solely based on frequency domain information is challenging.

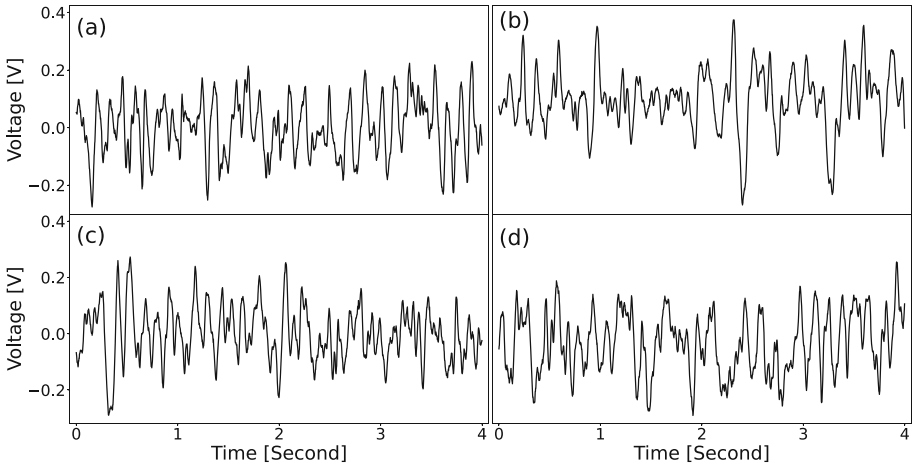


Fig. 1. A typical example of EEG time series for REM (a), NREM (b), HT (c) and LT (d).

2.2 Quantile Graph

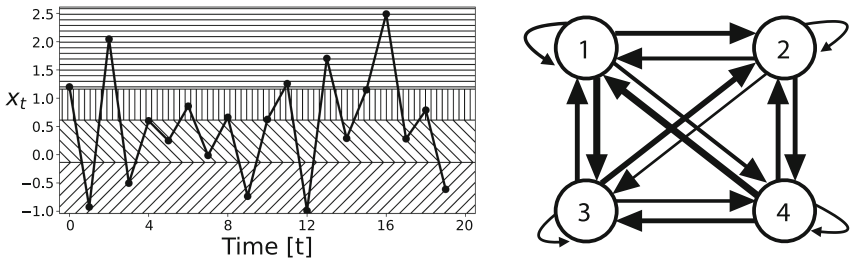


Fig. 2. An example of a Quantile Graph (right) which is generated from a time series (left), where the number of bins is set to 4.

The process begins by ranking the time series values, after which each value is assigned to its corresponding quantile. The network is then formed based on the transitions between these quantile values, which serve as the nodes. Connections between nodes are created by tracking the transitions over time, with the weights of the edges representing the frequency or intensity of these transitions. Both the weights and directions of the edges are determined by the movements between different quantile values in the time series. The weight from the ith quantile to the jth quantile is stored in the (i, j) element of the adjacency matrix. For example, consider a time series data $\{10, 7, 2, 72, 8, 94, 1, 19, 11, 12\}$. Dividing this data into five quantiles results in the transformed sequence $\{3, 2, 1, 5, 2, 5, 1, 4, 3, 4\}$. From

this transformed time series, an adjacency matrix is then constructed, where each node corresponds to a specific quantile value.

Figure 2 is an example of a Quantile Graph. The adjacency matrix can be treated as a transition matrix by normalizing its weight. The network constructed by the Quantile Graph is directed and weighted between the nodes. To account for the time scale of transitions in the data, a time delay parameter τ is introduced, resulting in the transformation of the original time series$\{x_i\}_{i=0,1,2,\ldots}$ into a new series $\{x_{k+i\tau}\}_{i=0,1,2,\ldots}$. Introducing a time delay effectively filters out transitions within a time span shorter than τ, allowing for the analysis of more distant data points. The quantile magnitude determines the precision of these transitions. By excluding the influence of short-term changes and adjusting the quantile to enhance resolution, an optimal temporal resolution can be achieved from a broader perspective. This study uses quantiles of 4, 10, 20, 50, and 80, with time delays ranging from 1 to 79.

2.3 Features

In this research, we analyze not only statistics, including the mean, standard deviation, minimum, maximum, and autocorrelation functions, but also network features derived from the Quantile Graph as characteristics of the time series. We expect that the classification of sleep stages incorporating features from Quantile Graphs will outperform the classification based solely on basic statistics. To validate this expectation, we compare two sets of features: one that consists exclusively of basic statistics and another that includes features derived from Quantile Graphs. We anticipate that the latter will demonstrate superior performance due to its incorporation of additional information beyond frequency.

The network features used in this study are the average clustering coefficient, average shortest path length, modularity, and Shannon entropy. The network features can be calculated in 4 distinct ways based on the presence or absence of weights and directionality. Consequently, the features were computed using these 4 methods, except for modularity in the case of directed graphs. Shannon entropy H_1 is calculated based on the transition matrix derived from the Quantile Graph:

$$H_1 = -\sum_{i,j=1}^{\mathrm{N}} p_i^* p(i \to j) \log p(i \to j), \tag{1}$$

where N, p_i^* and $p(i \to j)$ denote total number of quantiles (nodes), ith element of the stationary distribution, and the (i, j) element of the transition matrix, respectively. We compute H_1 in 2 different ways. (1) We empirically calculate the stationary distribution p_i^* from the data. (2) By utilizing the eigenvector of the transition matrix, we theoretically computed the stationary distribution. In addition, assuming the 2nd order Markov chain, we computed the Shannon

entropy using the following equation:

$$H_2 = -\sum_{i,j,k=1}^{\mathrm{N}} p^*_{ij} p(ij \to jk) \log p(ij \to jk), \tag{2}$$

where p^*_{ij} is the element of the stationary distribution at (i, j) and $p(ij \to jk)$ denotes the transition probability from i to k via j. For the 2nd order case, we empirically computed the stationary distribution p^*_{ij} from the data. If the transitions satisfies a 1st order Markov process, then the values of H_1 and H_2 should be equal. The network features were calculated by varying the quantiles and time delay.

2.4 Model and Metrics

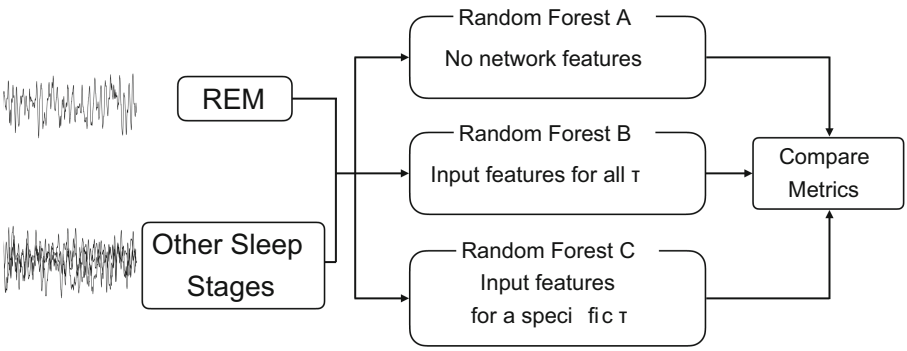


Fig. 3. Analysis procedure for REM

The classification model employed in this study is a random forest (Fig. 3), where each model is designed to classify whether a sample belongs to a specific sleep stage or to other stages. For instance, one model classifies REM sleep while grouping the other stages, which are NREM, HT, and LT, together. For each sleep stage classification, we examined 3 types of input data. Random forest A uses only basic statistics as input, without incorporating any network features. In contrast, random forest B combines basic statistics with network features for all values of τ, whereas random forest C includes basic statistics along with network features for specific values of τ. We set the depth of the trees to 4, and the input dimensions are 133, 5268 and 198 for random forest A, B, and C, respectively. The network features are calculated using varying parameters derived from quantiles. The models are evaluated 50 times by varying the training and validation datasets, allowing us to compare the evaluated scores for each model and analyze the impact of the network features on classification performance. For each learning and evaluation process, 80% of the entire dataset is used as

the training set, while the remaining 20% is used as the validation set, with the training and validation sets randomly shuffled.

The contribution of each feature to the prediction is evaluated by its relative importance. It is important to note that random oversampling is employed during the training process due to an imbalanced class ratio. If the class ratio is significantly imbalanced, the model will predict a specific class frequently. To avoid this issue, a random sampling approach is employed to ensure the equality of the data sets. Thus, the class ratio in the training data sets is balanced, whereas that in the validation data sets remains imbalanced. The results are evaluated by accuracy, macro recall, macro F1 score, Cohen's Kappa coefficient κ, and Matthews correlation coefficient MCC [5]. These coefficients are defined by

$$\kappa = \frac{\sum_{l \in \text{class}} \text{TP}_l - \sum_{l \in \text{class}} \text{TP}_l \cdot \text{FP}_l}{\text{N}^2 - \sum_{l \in \text{class}} \text{TP}_l \cdot \text{FP}_l}, \tag{3}$$

$$\text{MCC} = \frac{(\text{TP} + \text{TN}) \cdot N - \sum_{l \in \text{class}} (TP_l + FP_l) \cdot TP_l}{\sqrt{(N^2 - \sum_{l \in \text{class}} (TP_l + FP_l)^2)(N^2 - \sum_{l \in \text{class}} TP_l^2)}}, \tag{4}$$

where TP, TN, FP, and FN denote True Positive, True Negative, False Positive and False Negative, respectively. In the equations, class represents REM, NREM, HT and LT. We note that macro scores are employed to mitigate the effect of data imbalance across sleep stages.

3 Results

The metrics obtained from classifying each sleep stage using random forest A and B are presented in Fig. 4. For all 5 metrics, random forest B tends to exhibit higher values than random forest A for REM sleep, while showing lower values for NREM, HT, and LT. This suggests that the characteristics of EEG during REM are influenced not only by basic statistics but also by different aspects of information. The NREM score is already sufficiently high, as it is characterized well by its amplitude. Consequently, the inclusion of network features has a limited effect on classification and may even negatively impact the scores. The performance also declines in HT and LT. Thus, it can be inferred that the network features are not sufficiently correlated with the classification of these sleep stages. Additionally, the reduced metrics may be due to the relatively small data size compared to the number of input dimensions and the insufficient depth of the random forest trees. Figure 5 shows the normalized importance of the top 50 features for each sleep stage. For all sleep stages, autocorrelation functions are shown to have high importance, indicating that frequency information is useful for classification. For REM and HT, the autocorrelations at time delays of 33 and 68 correspond to characteristic frequencies of 16 Hz and 7–8 Hz, respectively, which are shown to have high importance. Standard deviation plays a particularly significant role in classifying NREM and arousal, emphasizing the importance of amplitude in these stages. Markov features also show relatively

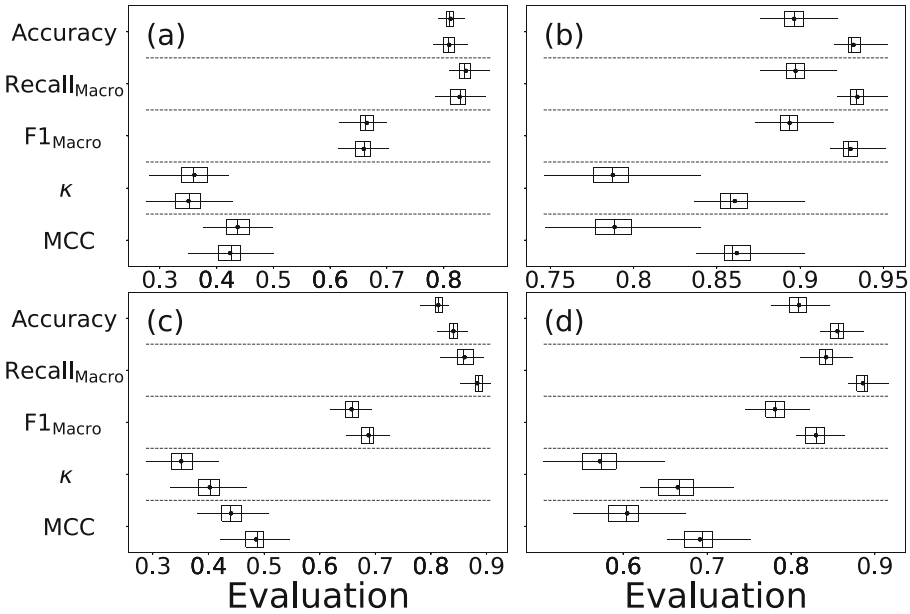


Fig. 4. Box plots for each metric of random forest A (lower) and random forest B (upper) for REM (a), NREM (b), HT (c) and LT (d).

high importance in classifying LT, suggesting that while frequency-based classification of LT is challenging, network features may provide potential for improved classification.

The mean evaluated metrics from classifying each sleep stage using random forests A and C are shown in Fig. 6. The metrics for random forest C generally show higher values compared to those for random forest B (Fig. 4), indicating that in our current model, using features from specific values of τ improves performance more than using features from all values of τ. When comparing the metrics of random forest C with those of random forest A, there is little difference observed for HT and LT, but performance improvements are evident for REM and NREM. In fact, a t-test conducted on the mean values for REM and NREM shows statistically significant improvements in performance for smaller values of τ. Looking at the overall trend across the 5 metrics for random forests C, REM tends to reach its maximum performance around $\tau = 12$, while NREM exhibits a monotonic decrease in performance as τ increases, with random forests C and A performing similarly around $\tau = 20$. Figure 7 displays the normalized importance of the top 50 features for each sleep stage. Unlike the results from random forest B (Fig. 5), which showed lower importance values, several features across all sleep stages now exceed 0.05, emphasizing the varying contributions of the input features more clearly. For NREM, HT, and LT, basic statistics maintain high importance values, consistent with previous findings. However, in the case of REM, we observe a notable difference: features such as average

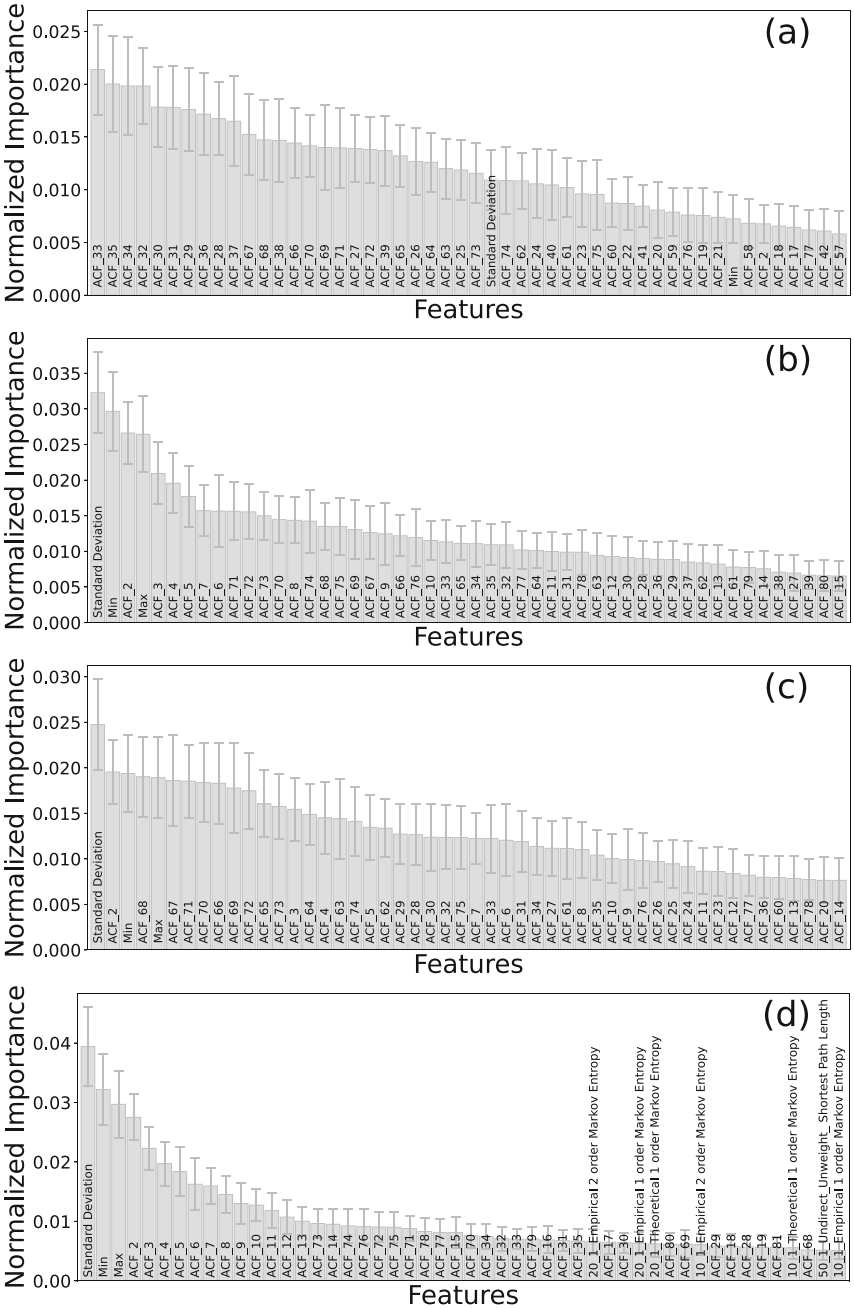


Fig. 5. Mean of normalized importances of random forest B for REM (a), NREM (b), HT (c) and LT (d). A feature ACF_n indicates autocorrelation of time delay n. For example, q_τ_Empirical 2 order Markov Entropy represents H_2 at q-th quantile and time delay τ. Error bars represent the standard deviation.

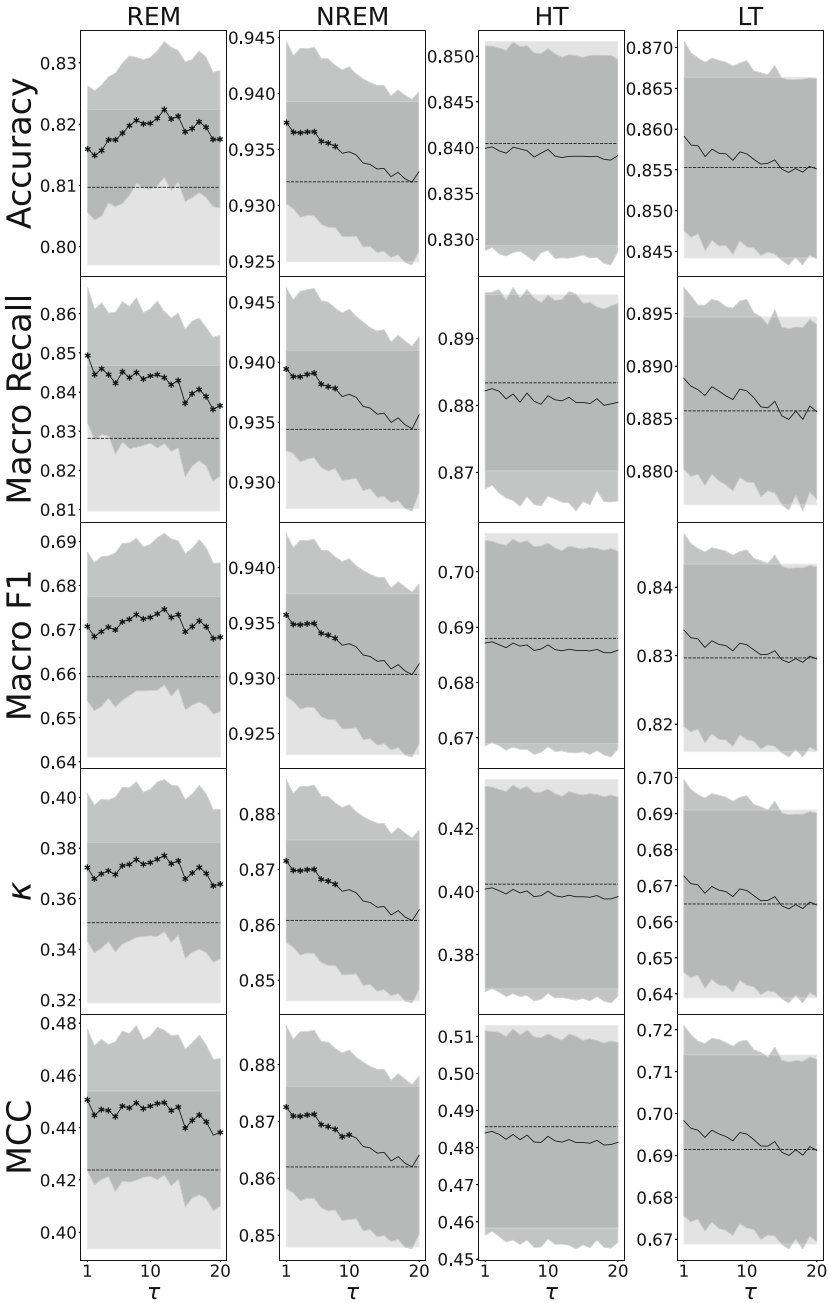


Fig. 6. Mean evaluated scores of random forest A (dotted line) and random forest C (solid line) as a function of time delay τ for accuracy (a), macro recall (b), macro F1 score (c), Cohen's Kappa coefficient (d), and Matthews correlation coefficient (e). * indicates the significant difference ($p < 0.05$) as determined by a t–test at the τ. Gray areas show standard deviation.

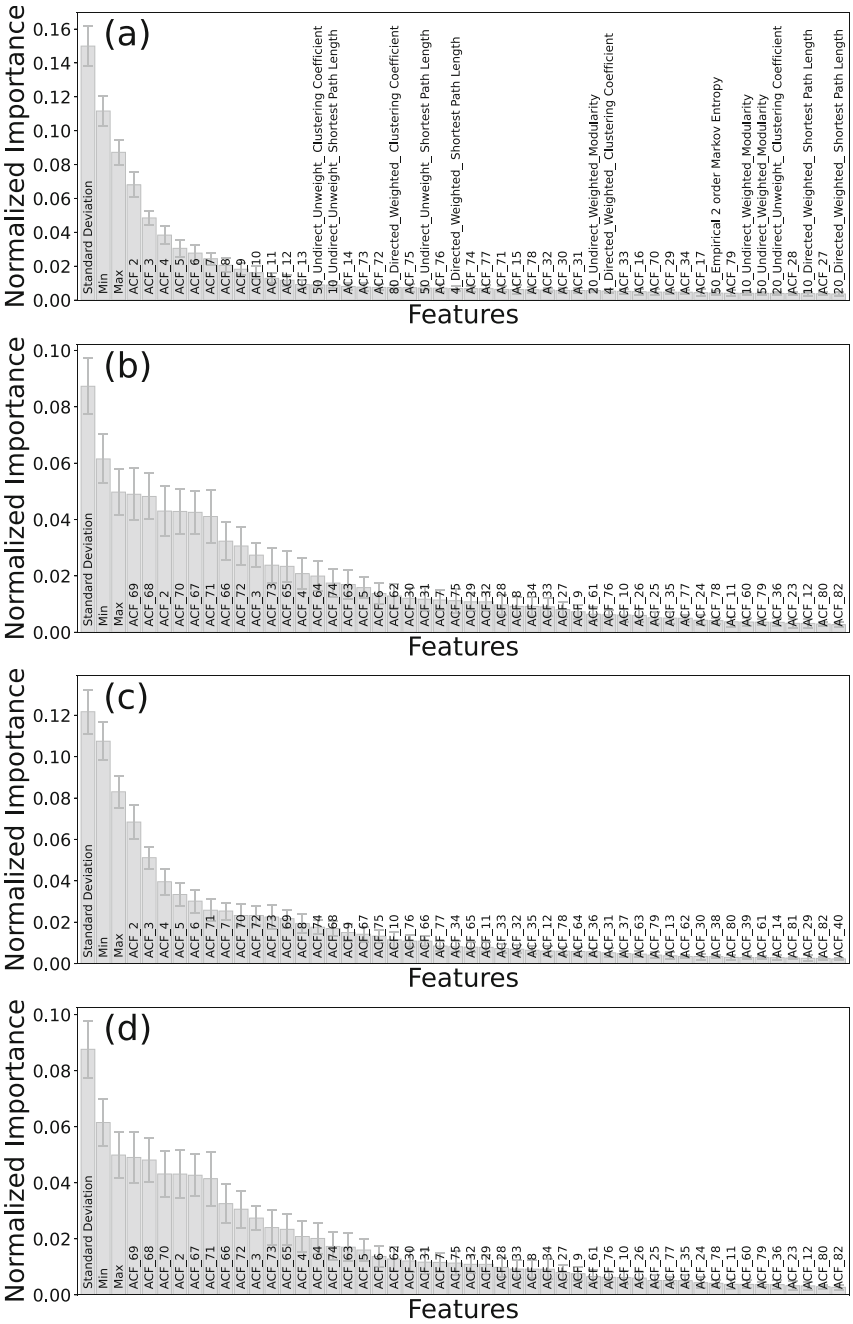


Fig. 7. Mean of normalized importances of random forest C for REM at $\tau = 12$ (a), NREM at $\tau = 1$ (b), HT at $\tau = 2$ (c), and LT at $\tau = 1$ (d). Error bars represent the standard deviation.

clustering coefficient, average shortest path length, modularity, and H_2 exhibit high importance values as well. This suggests that, in addition to improved performance, the effectiveness of network features in REM classification is further validated by their significant importance values.

4 Conclusion

We evaluated the classification performance of EEG time series networks using random forests, comparing various metrics to assess their effectiveness. These networks are constructed using the Quantile Graph. The selection of the parameters, such as the quantile value and the time delay τ, is critical in achieving enhanced classification performance. The optimal network features derived from these parameters capture both the periodicity and nonstationary changes in the time series, leading to improved classification accuracy. Our findings demonstrate the superiority of network features; the average clustering coefficient, average shortest path length and modularity, over entropy computed from transition matrices. Specifically, the average clustering coefficient and average shortest path length show significant contributions to the classification of REM sleep. However, the network features are less effective for classifying other sleep stages compared to their impact on REM sleep classification. The selection of an optimal time delay is crucial for improving classification across each sleep stage.

References

1. Ahmadlou, M., Adeli, H., Adeli, A.: New diagnostic EEG markers of the Alzheimer's disease using visibility graph. J. Neural Transm. **117**, 1099–1109 (2010)
2. Aithal, N., Pradeep, C.S., Sinha, N.: MCI detection using fMRI time series embeddings of recurrence plots. arXiv preprint arXiv:2311.18265 (2023)
3. Campanharo, A.S., Doescher, E., Ramos, F.M.: Application of quantile graphs to the automated analysis of EEG signals. Neural Process. Lett. **52**, 5–20 (2020)
4. Campanharo, A.S., Sirer, M.I., Malmgren, R.D., Ramos, F.M., Amaral, L.A.N.: Duality between time series and networks. PLoS ONE **6**(8), e23378 (2011)
5. Gorodkin, J.: Comparing two K-category assignments by a K-category correlation coefficient. Comput. Biol. Chem. **28**(5–6), 367–374 (2004)
6. Jie, L., Hao, W., Huibin, X., Shengqiao, B., Li, L.: A modified Markov transition probability-based network constructing method and its application on nonlinear time series analysis. In: 2020 Chinese Control And Decision Conference (CCDC), pp. 4868–4872 (2020)
7. Kobayashi, T., Madokoro, S., Wada, Y., Misaki, K., Nakagawa, H.: Human sleep EEG analysis using the correlation dimension. Clin. Electroencephalogr. **32**(3), 112–118 (2001)
8. Koyama, K., Ito, M.I., Ohnishi, T.: Fluctuation in grocery sales by brand: an analysis using Taylor's law. Rev. Socionetwork Strat. **16**, 417–430 (2022)
9. Koyama, K., Sakaguchi, M., Ohnishi, T.: A noise-robust approach to estimate dimension of sleep EEG in mice using permutation entropy. In: 2024 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1–8 (2024)

10. Lacasa, L., Luque, B., Ballesteros, F., Luque, J., Nuno, J.C.: From time series to complex networks: the visibility graph. Proc. Natl. Acad. Sci. **105**(13), 4972–4975 (2008)
11. Lee, M.: Early warning detection of thermoacoustic instability using three-dimensional complexity-entropy causality space. Exp. Thermal Fluid Sci. **130**, 110517 (2022)
12. Liu, L., Wang, Z.: encoding temporal Markov dynamics in graph for visualizing and mining time series. In: Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence, pp. 178–184 (2018)
13. Luque, B., Lacasa, L., Ballesteros, F., Luque, J.: Horizontal visibility graphs: exact results for random time series. Phys. Rev. E-Stat. Nonlinear Soft Matter Phys. **80**(4), 046103 (2009)
14. Mohammadpoory, Z., Nasrolahzadeh, M., Haddadnia, J.: Epileptic seizure detection in EEGs signals based on the weighted visibility graph entropy. Seizure **50**, 202–208 (2017)
15. Osmane, A., Dimmock, A.P., Pulkkinen, T.I.: Jensen-Shannon complexity and permutation entropy analysis of geomagnetic auroral currents. J. Geophys. Res. Space Physics **124**(4), 2541–2551 (2019)
16. Pineda, A.M., Ramos, F.M., Betting, L.E., Campanharo, A.S.: Quantile graphs for EEG-based diagnosis of Alzheimer's disease. PLoS ONE **15**(6), e0231169 (2020)
17. Shirazi, A., Jafari, G.R., Davoudi, J., Peinke, J., Tabar, M.R.R., Sahimi, M.: Mapping stochastic processes onto complex networks. J. Stat. Mech. Theory Exp. **2009**(07), P07046 (2009)
18. Silva, V.F., Silva, M.E., Ribeiro, P., Silva, F.: Time series analysis via network science: concepts and algorithms. Wiley Interdisc. Rev. Data Min. Knowl. Discov. **11**(e1404), 1–39 (2021)
19. Silva, V.F., Silva, M.E., Ribeiro, P., Silva, F.: Novel features for time series analysis: a complex networks approach. Data Min. Knowl. Disc. **36**, 1062–1101 (2022)
20. Supriya, S., Siuly, S., Wang, H., Zhang, Y.: EEG sleep stages analysis and classification based on weighed complex network features. IEEE Trans. Emerg. Topics Comput. Intell. **5**(2), 236–246 (2021)
21. Zhu, G., Li, Y., Wen, P.: Analysis and classification of sleep stages based on difference visibility graphs from a single-channel EEG signal. IEEE J. Biomed. Health Inf. **18**(6), 1813–1821 (2014)

A Fully-Dynamic Approximation Algorithm for Maximum Weight *b*-Matchings in Graphs

Fabian Brandt-Tumescheit(✉), Frieda Gerharz, and Henning Meyerhenke

Humboldt-Universität zu Berlin, Berlin, Germany
{brandtfa,meyerhenke}@hu-berlin.de

Abstract. Matching nodes in a graph $G = (V, E)$ is a well-studied algorithmic problem with many applications. The b-matching problem is a generalization that allows to match a node with up to b neighbors. This allows more flexible connectivity patterns whenever vertices may have multiple associations. The algorithm B-SUITOR [Khan et al., SISC 2016] is able to compute a (1/2)-approximation of a maximum weight b-matching in $\mathcal{O}(|E|)$ time. Since real-world graphs often change over time, fast dynamic methods for b-matching optimization are desirable.

In this work, we propose DYN-B-SUITOR, a dynamic algorithm for the weighted b-matching problem. As a non-trivial extension to the dynamic Suitor algorithm for 1-matchings [Angriman et al., JEA 2022], our approach computes (1/2)-approximate b-matchings by identifying and updating affected vertices without static recomputation. Our proposed algorithm is fully-dynamic, i. e., it supports both edge insertions and deletions, and we prove that it computes the same solution as its static counterpart.

In extensive experiments on real-world benchmark graphs and generated instances, our dynamic algorithm yields significant savings compared to the sequential B-SUITOR, e. g., for batch updates with 10^3 edges with an average acceleration factor of $10^3\times$. When comparing our sequential dynamic algorithm with the parallel (static) B-SUITOR on a 128-core machine, our dynamic algorithm is still $59\times$ to $10^4\times$ faster.

Keywords: approximate b-matching · dynamic graph algorithm · algorithmic network analysis

1 Introduction

A b-matching M in a graph $G = (V, E)$ is a subgraph of G such that each vertex is incident to at most b edges in M. This way, the concept of b-matchings generalizes the widely known notion of matchings, which are 1-matchings ($b = 1$). b-matchings ($b \geq 1$) find applications in various fields, including social network anlaysis [22], network design [7,28], scheduling [34], resource allocation [12,14,19], and fraud detection in finance networks [29]. Compared to

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 139–151, 2025.
https://doi.org/10.1007/978-3-031-82439-5_12

ordinary matchings, b-matchings ($b \geq 2$) allow more flexible connectivity patterns in scenarios where vertices may have multiple (yet still a restricted number of) associations.

Graphs derived from real-world applications are often dynamic in nature; consider social, information, or infrastructure networks, for example. While adding friends, comments, or likes in social networks is just one click of work, the update of a matching can become comparably time-consuming in a large graph when it is recomputed entirely from scratch. To avoid a bottleneck when updating M in large graphs, one needs to restrict the work to those graph areas that are affected by the changes in G, while ensuring that the new b-matching reflects these changes as well. When supporting both edge insertions and deletions in this manner, we refer to an algorithm as fully-dynamic. For the 1-matching problem, fully-dynamic algorithms have been proposed and implemented previously for maximum weight matching (i. e., a set of independent edges with maximum total weight) [2,5,6,8,9,21]. The algorithm proposed by Angriman et al. [2] is based on the static Suitor algorithm [31]. Since B-SUITOR by Khan et al. [24] shares the same foundation, it is also a natural starting point for the design of a fully-dynamic b-Suitor algorithm for b-matchings. While there exists other work on dynamic b-matching algorithms, the approaches are either theoretical in nature [6,11,17] and have not been thoroughly studied in practice or they support only edge insertions [12].

Contributions. In this paper, we propose DYN-B-SUITOR – a fully-dynamic b-matching algorithm that maintains a (1/2)-approximation of a maximum weight b-matching (see Sect. 3). The algorithm is influenced mainly by two previous works: B-SUITOR by Khan et al. [24] (Sect. 2.3) and the dynamic Suitor algorithm for the 1-matching problem [2] (Sect. 2.4). Starting initially with a b-matching derived from B-SUITOR, our fully-dynamic algorithm finds and updates all vertices whose matching decision is potentially affected by the recent changes to the graph. Our main theoretical result (Theorem 1), which requires a series of lemmas, is to prove that DYN-B-SUITOR computes the same result as B-SUITOR.

According to our extensive experiments (Sect. 4), our DYN-B-SUITOR implementation yields an algorithmic speedup (= static running time divided by dynamic one) of 10^6 over static recomputation for single updates (edge insertion/deletion) and 10^3 for batches with 10^3 edges. Moreover, for all 20 networks from four different categories (social, infrastructure, sparse, generated), the ratio of algorithmic speedup to batch size is slightly above 1. This advantage of batch processing is due to synergies when visiting and updating affected nodes and edges from the same batch. Running times are in the millisecond range or below, making the algorithm $59\times$ to $10^4\times$ faster than the state-of-the-art parallel B-SUITOR implementation [24] on 128 CPU cores, depending on the batch size.

The appendix of this paper is omitted due to space constraints; it can be found in the extended version of this paper [1].

2 Preliminaries

2.1 Problem Definition and Notation

We consider undirected weighted graphs $G = (V, E, w)$ with $n := |V|$ vertices (= nodes), $m := |E|$ edges, and the edge weight function $\omega : E \to \mathbb{R}_{>0}$. The set of neighbors of a vertex u is denoted by $N(u)$ and the (unweighted) degree of u as $\deg(u) := |N(u)|$. We use a common tie-breaking rule to define a total ordering: in case the weight of two edges $\{u, t\}$ and $\{u, v\}$ is equal, we say that $\omega(u, t) > \omega(u, v)$ if $t < v$, i. e., if the node ID of t is smaller than the ID of v. This is required for breaking ties when selecting the heaviest neighbor of u.

A b-matching is a set of edges $M \subseteq E$ in which no more than $b(u)$ edges are incident to u, where $b(\cdot)$ is a function that maps every vertex u to a natural number. Thus, b-matchings are a generalization of ordinary matchings; the latter are a special case by using the constant function $b : u \to 1$. The weight of a b-matching M is defined as $\omega(M) = \sum_{e \in M} \omega(e)$. If M has maximum weight among all b-matchings, it is called a maximum weight b-matching (MWBM). This paper deals with finding an approximate MWBM. An edge $\{u, v\}$ is called *matched* if it belongs to M; otherwise it is called *unmatched*. If a vertex u is incident to b edges from M, then the vertex is called *saturated*. If not, it is *unsaturated*. In general, different vertices can have different saturation goals $b(\cdot)$. In our setting, these goals do not change when the graph changes over time.

2.2 Related Work

The MWBM problem has been studied both in terms of static and dynamic settings. We give an overview over both; afterwards, we explain two algorithms with high relevance for this paper in more detail, B-SUITOR for b-matchings [24] and the dynamic Suitor algorithm for 1-matchings [2]. For the dynamic 1-matching problem, a wide spectrum of algorithms have been proposed in the literature; we refer the interested reader to Ref. [5] for an overview.

Exact algorithms for MWBM have been around since the 1960s. Overall, the time complexity for computing the exact MWBM solution using known techniques is $\mathcal{O}(|V||E|)$ [18], which can become impractical for very large instances. As a result, several approximation algorithms have been proposed.

Huang and Pettie [23] proposed a $(1 - \epsilon)$ approximation algorithm for the MWBM problem by approximating the conditions for a relaxed complementary slackness formulation of the original problem. While the running time is nearly linear in $|E|$ using a scaling framework [16] for sufficiently large values of ϵ, the algorithm itself is theoretical in nature. Subsequently, Samineni et al. [36] addressed the MWBM problem by formulating an LP-relaxation based on dual variables. Compared to Huang and Pettie, their approach is more feasible for an actual implementation but introduces an additional dependency on b in the time complexity. However, the algorithm is only suitable for bipartite graphs, where vertices are split in *bidders* and *objects* in order to define an auction-like setup.

Another set of approximation algorithms is based on local optimization. This includes a $(2/3 - \epsilon)$ randomized algorithm [32] that picks vertices uniformly at

random and tries to match a neighbor based on the maximum gain between matched and unmatched neighbors. Its running time depends on $\log(1/\epsilon)$, b and the number of edges; moreover, due to its randomized nature, its result is not deterministic. Several (1/2)-approximation algorithms are based on the GREEDY algorithm by Mestre [32]. They remove the running time dependency on ϵ and produce identical b-matchings [15,24,35]. The b-suitor algorithm by Khan et al. [24] proves to be the fastest technique within the family of greedy algorithms. According to empirical results [24], the solution quality can be much better than the proven (1/2)-approximation, with a margin of only $\sim 1\%$ from the optimal value for some real-world networks. The algorithmic construction makes the b-suitor algorithm also suitable for parallelization. While the original publication already includes a parallel variant for shared memory [24], more recent publications adapt the construction to GPUs [33] and distributed environments [25]. Since our dynamic algorithm uses the b-suitor algorithm initially and computes the same matchings, we explain the latter in more detail in Sect. 2.3.

Concerning dynamic algorithms for MWBM on general graphs, the contributions so far have been theoretical in nature [6,11,17]. Also for bipartite graphs, existing algorithms have yet to be implemented [10,37]. One exception is a reinforcement learning approach by Wang et al. [38], which is concerned with inserting single nodes and corresponding edges into the (bipartite) graph together at the same time. Its idea is to use small batches of inserted node-edge tuples together and do a greedy optmization locally inside the batch. The optimal batch splitting is then learned from the Hungarian algorithm. So far, however, there is no way known to transfer this approach to MWBM on general graphs. To the best of our knowledge, our proposed algorithm is the first implemented fully-dynamic algorithm for the (approximate) MWBM problem for general graphs.

2.3 The B-SUITOR Algorithm

B-SUITOR by Khan et al. [24] is a (1/2)-approximation algorithm; it extends the Suitor algorithm [31] for matchings to $b \geq 2$. In its core, B-SUITOR iterates over all nodes u of G and tries to find up to $b(u)$ matching partners for u. Conceptually, each node negotiates with its neighbors to form a shared proposal, referred to as suitors. To this end, for each node there exists a min-priority queue $S(u)$ with a maximum size of $b(u)$, representing all neighbors of u which have already proposed to u as potential suitors. If $S(u)$ contains $b(u)$ elements, we call it saturated and $S(u).min$ returns the suitor with the lowest edge weight. In case $|S(u)| < b(u)$, $S(u)$ is unsaturated and the result of $S(u).min$ is `None`.

While the original pseudocode includes an additional helper set $T(u)$ that contains intermediate matches until the algorithm finishes, adaptations [25,33] remove this set. In this adapted version, the invariant $v \in S(u) \Leftrightarrow u \in S(v)$ is maintained throughout the execution of the algorithm. In the following, we refer to this invariant as the *S-invariant* for convenience.

When the main vertex of the current iteration is u, the goal is to find a matching partner p_u for u according to:

$$p_u = \underset{v \in N(u) \setminus S(u)}{\operatorname{argmax}} \{\omega(u,v) \mid \omega(u,v) > \omega(v, S(v).min)\} \tag{1}$$

If a matching partner of this kind is found, u updates its list of suitors $S(u)$ and also adds itself to the suitor list of that partner. If that partner was saturated before, we need to remove its previous minimum suitor and find a new partner for the removed vertex recursively (also see the iterative version as Algorithm 2 in Appendix A). The key result about B-SUITOR is:

Proposition 1. ([24]). *The S-queues of all nodes form a valid b-matching $M = \{\{u,v\} \in E \mid u \in S(v), v \in S(u)\}$. M is a (1/2)-approximation of a (not necessarily unique) maximum weighted b-matching.*

The overall running time of B-SUITOR is $\mathcal{O}(m\Delta \log \beta)$, with Δ as maximum vertex degree of G and $\beta = \max_{v \in V} b(v)$. Obviously, by setting $b = 1$, B-SUITOR can also be used for solving the corresponding 1-matching problem. Khan et al. [24, Sec. 3] also design a parallel version of the algorithm for shared-memory systems. It uses a global queue and vertex-specific locks whenever $S(u)$ is updated. Since our dynamic algorithm is sequential in nature, we mainly compare against the sequential B-SUITOR. Note that our experiments also show that DYN-B-SUITOR is much faster than recomputing from scratch in parallel.

2.4 The Dynamic Suitor Algorithm

For the 1-matching problem, Angriman et al. [2] design a fully-dynamic algorithm based on the Suitor algorithm by Manne et al. [31]. Starting with a matching M in the initial graph, e. g., computed by the static Suitor algorithm, the main idea of the dynamic matching (DM) algorithm is to update the matching by identifying all vertices influenced by the changes in the graph. For this to work, Angriman et al. introduce the definition of *affected* vertices [2, Def. 1]. The main idea is to call any vertex u affected for which there exists a better suitor in the neighborhood $N(u)$. Since $b = 1$, $S(u)$ now refers to a single suitor instead of a list. As a result, affected vertices can be found iteratively by repeatedly applying Eq. (2) – a variation of Eq. (1):

$$s_u = \underset{v \in N(u) \setminus S(u)}{\operatorname{argmax}} \{\omega(u,v) \mid \omega(u,v) > \omega(v, S(v))\} \tag{2}$$

Whenever s_u is not `None`, u is affected and we connect u and s_u as new suitors. The iterative process continues by checking Eq. (2) again for the previous suitor of s_u. This process continues until no new affected vertex is found (i. e., Eq. (2) returns `None` for the vertex in question).

The iterative pursuit of locally affected vertices traverses update paths for both edge insertions and deletions. Each path starts at one endpoint u or v, respectively, of the updated edge $\{u,v\}$. Angriman et al. prove that after handling all vertices on the path, the resulting matching equals the matching computed by the static Suitor algorithm. In the worst-case, this iterative process

takes $\mathcal{O}(n+m)$ time. Especially for single edge updates, only a fraction of nodes is affected, though. Even for batch updates with up to 10^3 nodes, the algorithmic speedup (DM vs Suitor) is still several orders of magnitude.

Note that the rules for identifying affected vertices in their FindAffected routine [2, Alg. 1] cannot be applied or directly translated to the b-matching case, since nodes might need to be updated multiple times if $b > 1$. Moreover, the dynamic Suitor algorithm traverses the update paths twice, leaving room for improvement.

3 The Dyn-b-suitor Algorithm

The core ideas of our dynamic algorithm Dyn-b-suitor extend and generalize the sequential b-suitor as well as the DM framework (Sects. 2.3 and 2.4). The resulting algorithm computes a (1/2)-approximate maximum weight b-matching for dynamic graphs by revising vertices affected by an edge update. It can handle single edges and batches of edges to be inserted or removed.

Due to overlapping design decisions, we reuse notation from Angriman et al. [2]. For the different states of the graph and our data structures (i) before, (ii) during, and (iii) after the manipulation of the graph and the execution of the algorithm, we use the notation in Table 1. Since the dynamic algorithm supports both edge insertions and deletions, $\widetilde{G} = (V, \widetilde{E})$, and $\widetilde{N}(u)$ denote the state of the graph data structure after edges have been added/removed. $S/\widetilde{S}$ follows the definitions from Sect. 2.3, representing the data structures for the suitors w. r. t. the original/updated graph. For Dyn-b-suitor, we introduce subscripts similar to the dynamic Suitor algorithm. They refer to state $^{(i)}$ (intermediate) directly after the graph edge update (addition or removal) and $^{(f)}$ (final) after the dynamic algorithm has been run. The intermediate b-matching $M^{(i)}$ considers the updated graph data structure; in this state, it may differ from the matching $\widetilde{M}$ obtained by the static algorithm. The deviation of $M^{(i)}$ with respect to $\widetilde{M}$ could either be caused by an edge insertion or removal and the associated change of suitor lists.

The goal of the dynamic algorithm is to efficiently update $S^{(i)}(u)$ after an edge insertion or removal in order to arrive at a final state $S^{(f)}(u)$, so that the invariant $v \in S^{(f)}(u) \Leftrightarrow u \in S^{(f)}(v)$ holds and also that the resulting $M^{(f)}$ equals the b-matching $\widetilde{M}$ computed by b-suitor. To achieve this goal, we define the notion of a *locally affected* vertex, extending Eq. (1):

Table 1. Notation used for dynamic maximum weight b-matching computations.

state	graph	neighbors	suitors	b-matching
original graph + after b-suitor	$G = (V, E)$	$N(u)$	$S(u)$	M
updated graph + after b-suitor	$\widetilde{G} = (V, \widetilde{E})$	$\widetilde{N}(u)$	$\widetilde{S}(u)$	$\widetilde{M}$
updated graph + <u>during</u> Dyn-b-suitor	$\widetilde{G} = (V, \widetilde{E})$	$\widetilde{N}(u)$	$S^{(i)}(u)$	$M^{(i)}$
updated graph + <u>after</u> Dyn-b-suitor	$\widetilde{G} = (V, \widetilde{E})$	$\widetilde{N}(u)$	$S^{(f)}(u)$	$M^{(f)}$

Definition 1. *A vertex $u \in V$ is called locally affected iff there exists a better suitor in the updated neighborhood $\widetilde{N}(u)$, i. e., for $s_u = \operatorname{argmax}_{v \in \widetilde{N}(u) \setminus S^{(i)}(u)} \{\omega(u,v) \mid \omega(u,v) > \omega\left(v, S^{(i)}(v).min\right)\}$ it holds that $s(u) \neq$ `None` $\wedge\, \omega\left(u, S^{(i)}(u).min\right) < \omega(u, s_u)$.*

Algorithm 1. Function FindAffected

```
1: function FindAffected(x)
2:     done ← False
3:     cu ← x                                                    ▷ Init current node
4:     ca ← S^(i)(cu).min                                        ▷ Init candidate node
5:     looseEnd ← None
6:     repeat
7:         done ← True
8:         ca ← argmax_{v∈Ñ(cu)\S^(i)(cu)} {ω(cu,v):ω(cu,v)>ω(v,S^(i)(v).min)}
9:         if ca == None then                                    ▷ Def. 1 not fulfilled → stop here
10:            break
11:        end if
12:        prevCu ← S^(i)(cu).insert(ca)                         ▷ Returns replaced item
13:        prevCa ← S^(i)(ca).insert(cu)
14:        if prevCu ≠ None then                                 ▷ Detect special case
15:            S^(i)(prevCu).remove(cu)
16:            looseEnd ← prevCu
17:        end if
18:        if prevCa ≠ None then                                 ▷ Found new affected node
19:            S^(i)(prevCa).remove(ca)                          ▷ Maintain S-invariant
20:            cu ← prevCa                                       ▷ Set current to new affected node
21:            done ← False
22:        end if
23:        ca ← S^(i)(cu).min
24:        cw ← ω(cu, ca)
25:    until done                                                ▷ Repeat loop until path is finished
26:    if looseEnd != None then
27:        FindAffected(looseEnd)                                ▷ Repair special case
28:    end if
29: end function
```

3.1 Finding Locally Affected Vertices

After a single edge update, our dynamic algorithm computes $M^{(f)}$ by finding all locally affected vertices iteratively, starting with endpoints of the updated edge (see FindAffected as Algorithm 1). In that sense, the main loop (lines 6 to 25) reuses ideas from the dynamic suitor algorithm. The main difference in Dyn-b-suitor is the application of the S-invariant from b-suitor throughout the dynamic update, i. e., consequently updating $S^{(i)}$ for every update of node u and its new partner (lines 12+13). Maintaining the S-invariant also enables Dyn-b-suitor to remove the need for traversing the update paths twice, thereby reducing the workload with respect to the dynamic suitor algorithm.

FindAffected is called for each endpoint of an edge update, initialized as cu in the algorithm (line 3). The goal is then to find a suitable candidate ca (line 8). To this end, the unmatched neighborhood $\widetilde{N}(cu) \setminus S^{(i)}(cu)$ of cu is searched.

If no eligible candidate is found or the candidate does not fulfill Definition 1, FindAffected exits the loop (line 10). Otherwise, we continue and match *cu* and *ca*, maintaining the *S-invariant*. If *ca* is saturated, i. e., *prevCa* is assigned to a not `None` value (lines 13 and 18), *ca* will also be removed from *prevCa*. Since *prevCa* is not *saturated* anymore and therefore might also fulfill Definition 1, the next iteration of the main loop uses *prevCa* as the new *cu* vertex (line 20). Similarly to the DM framework from Ref. [2], FindAffected traverses a so-called update path of affected nodes. This path ends when either *ca* was not saturated before the update or no suitable *ca* can be found.

Compared to b-suitor, FindAffected directly updates both suitor queues of the affected node *cu* and the new partner *ca*. To compensate for the possibility of recurring node updates, we introduce the concept of a *loose end*. A loose end is identified whenever there is a locally affected node that is already saturated (line 14). After the iterative update path has been finished, this loose end is processed in a second recursive run of FindAffected.

For Dyn-b-suitor we additionally define two more functions: EdgeInsertion$(\cdot,\cdot)$ and EdgeRemoval$(\cdot,\cdot)$. Given an inserted edge $e = \{u, v\}$, EdgeInsertion(u,v) checks whether u and v are locally affected by the update, i. e., whether the new edge should be in the final matching $\widetilde{M}$. Analogously, EdgeRemoval(u,v) checks whether the removal of e influences the suitor lists of u and v and therefore might induce changes to $\widetilde{M}$. If an edge update indeed renders u and v locally affected, EdgeInsertion(u,v) and EdgeRemoval(u,v) update $S^{(i)}(u)$ and $S^{(i)}(v)$ and call FindAffected$(\cdot)$ for both u and v, respectively, in order to identify and update all locally affected vertices.

In addition to the single edge insertion and removal, the algorithm can also be generalized to support multiple edge updates $B = \{e_1, \ldots, e_i\}$. For both variants (removal, insertion) the main idea is to apply their single edge update counterpart in a loop on B and $\widetilde{G}$. $\widetilde{G}$ is the updated graph based on G with either multiple removed ($\widetilde{G} = (V, \widetilde{E} = E \setminus B)$) or added edges ($\widetilde{G} = (V, \widetilde{E} = E \cup B)$). Note that the result of one loop iteration (i. e., the insertion/removal of a single edge) is not the final matching $M^{(f)}$ anymore, but instead the intermediate result $M^{(i)}$. Once the loop over all edge updates is completed, $M^{(i)}$ becomes $M^{(f)} = \widetilde{M}$.

The main theoretical result regarding Dyn-b-suitor is that, after applying the update routines, the resulting matching $M^{(f)}$ indeed equals $\widetilde{M}$, the b-matching computed by b-suitor:

Theorem 1. *Let $\widetilde{M}$ be the matching computed using the* b-suitor *on $\widetilde{G}$ and $B = \{e_1, \ldots, e_i\}$ be an batch of edge updates on G. After* Dyn-b-suitor *is finished, $M^{(f)}$ equals $\widetilde{M}$. $\widetilde{M}$ is a (1/2)-approximation of the MWBM problem.*

Proof. Due to space constraints, we provide the proof, based on a series of auxiliary results in Appendices B.1 to B.3, in Appendix B.4.

4 Experiments

The implementation of the dynamic b-suitor algorithm Dyn-b-suitor is done in C++, using the open-source framework NetworKit [3]. We plan to integrate the code into the next NetworKit release after paper acceptance. [1]

Since Dyn-b-suitor is sequential per design, the main competitor for comparison is the sequential implementation of b-suitor. The main comparison is done by tracking the algorithmic speedup, i. e., the running time of one complete run of b-suitor divided by the running time of the update routine of Dyn-b-suitor after a certain number of edge updates. To minimize potential running time differences caused by different toolkits, we implement b-suitor in NetworKit, too.

4.1 Settings

All main experiments for the sequential comparison are executed on a machine with a dual-socket Intel Xeon 6126 processor with 12 cores and 192 GB RAM. The parallel b-suitor by Khan et al. [24] is executed on an AMD Epyc 9754 with 128 cores and 192 GB of RAM. To ensure reproducibility, the command line tool Simexpal [4] is used to automate the experiments.

The datasets include a wide variety of graphs that can be categorized into four different types: sparsely connected graphs (*Sparse*), infrastructure networks (*Infra*), social networks (*Social*) and randomly generated (*Random*) graphs. Overall, these categories contain 20 different networks, ranging from 10^4 to 10^7 nodes and 10^6 to 10^8 edges. The networks are also chosen to have different underlying properties regarding average node degree. See Appendix C for more detailed information about the datasets.

The batch size, the type of operation, and the value of b form the major axes of the experimental investigation. Our tests are conducted with batch sizes ranging from 1 to 10^3 edges. The saturation goal b is taken from the set $\{1, 3, 10\}$, alongside randomly generated values between 1 and 10, using a uniform distribution. For each cross product of said axes (e. g., graph = belgium, operation = insert, batch size = 100, $b = 3$), Dyn-b-suitor is repeated 50 times. We use violin plots for visualizing the experimental data distributions. Also, the correctness of the Dyn-b-suitor implementation was checked successfully by comparing its b-matchings with those of b-suitor on 10^6 random $G(n, p)$ graphs.

4.2 Evaluation

Since the results are similar for edge insertions and edge removal concerning the algorithmic speedup, we show in this section only the visualization of edge insertion and removal combined for batches of size 10^3, see Fig. 1. The dynamic algorithm yields an average algorithmic speedup of $4.3 \cdot 10^3$ over all categories.

[1] Code, reproducibility instructions, and experimental results are available on Github: https://github.com/hu-macsy/cn-2024-dynamic-b-suitor-experiments.

The best results are achieved for infrastructure networks with an average algorithmic speedup of $6.5 \cdot 10^3$ for single-edge updates, whereas the lowest is the *Sparse* category with $2.1 \cdot 10^3$. This is in line with how the algorithm works. The chosen infrastructure networks have the lowest average node degree. As a result, even for low values of b, B-SUITOR is more likely to revisit nodes.

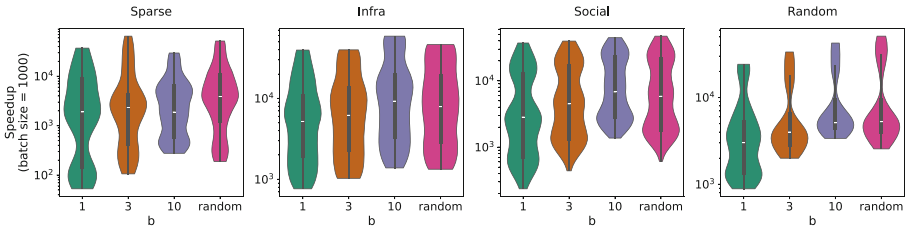


Fig. 1. Speedup for batches of size 10^3 (**edge insertion** and **edge removal** combined). For all analysed networks, the average speedup increases with the value of b.

In Appendix D we also break down the different operations into edge insertion and removal and give a more detailed overview of the results with different batch sizes. As expected, the maximum algorithmic speedup is achieved for single-edge updates: the combined average (geometric mean) algorithmic speedup is six orders of magnitude. When processing a batch of size $|B|$, one may expect a slowdown of $|B|$ w. r. t. to the running time for batch size 1. As can be seen from Figs. 7 and 8, the average slowdown for batch size 10^3 is 980, very close to expectation. This slight improvement can be explained by overlaps of the updates, i. e., a certain update from one edge is already covered by the update of a previous edge in the same batch.

Furthermore, the algorithmic speedup of DYN-B-SUITOR against the *parallel* implementation of B-SUITOR (on 128 cores) still shows average speedup of 59× for batch sizes of 10^3, see Appendix D.2.

For $b = 1$, i. e., an ordinary matching, we can compare our algorithmic speedups with those published in Angriman et al. [2] directly. Since DYN-B-SUITOR iterates over locally affected nodes only once, one might expect a measurable acceleration compared to the dynamic suitor algorithm. However, this is not the case; both algorithms yield the same order of magnitude. A reason for this deviation from expectation likely stems from line 8 in FINDAFFECTED: it involves more time-consuming checks compared to the dynamic Suitor algorithm.

5 Conclusions

Our experimental results show that the DYN-B-SUITOR algorithm outperforms a recomputation from scratch with the sequential B-SUITOR by three to six orders of magnitudes. Compared to the latter's parallel variant on 128 CPU cores, DYN-B-SUITOR is still at least 59× faster – depending on the batch size even

up to three orders of magnitudes. From a theoretical viewpoint, our work is a non-trivial extension of B-SUITOR, the dynamic Suitor algorithm, and their analysis. We could eventually prove that the resulting matching equals the one by the static algorithm and is thus a (1/2)-approximation. In practice, according to previous work, the weight of these b-matchings is often near the optimal one. Overall, our proposed algorithm provides a significant acceleration in all cases for updating b-matchings with high quality in a dynamic setting.

Acknowledgements. This work is partially supported by German Research Foundation (DFG) grant GR 5745/1-1 (DyANE).

References

1. Supplemental Material on Zenodo (2024). https://bit.ly/dynb
2. Angriman, E., Boroń, M., Meyerhenke, H.: A batch-dynamic suitor algorithm for approximating maximum weighted matching. ACM J. Experiment. Algorithmics (JEA) **27**(1), 1–41 (2022)
3. Angriman, E., van der Grinten, A., Hamann, M., Meyerhenke, H., Penschuck, M.: Algorithms for large-scale network analysis and the networkit toolkit. In: Bast, H., Korzen, C., Meyer, U., Penschuck, M. (eds.) Algorithms for Big Data: DFG Priority Program 1736, pp. 3–20. Springer Nature Switzerland, Cham (2022). https://doi.org/10.1007/978-3-031-21534-6_1
4. Angriman, E., et al.: Guidelines for experimental algorithmics: a case study in network analysis. Algorithms **12**(7), 127 (2019)
5. Angriman, E., Meyerhenke, H., Schulz, C., Uçar, B.: Fully-dynamic weighted matching approximation in practice. In: Michael, B., John, G., Blair, S.D. (eds.) SIAM Conference on Applied and Computational Discrete Algorithms (ACDA21), pp. 32–44. Society for Industrial and Applied Mathematics, Philadelphia, PA (2021). https://doi.org/10.1137/1.9781611976830.4
6. Azarmehr, A., Behnezhad, S., Roghani, M.: Fully dynamic matching: -approximation in polylog update time. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 3040–3061. SIAM (2024)
7. Bai, B., Li, W., Wang, L., Zhang, G.: Coded caching in fog-ran: b-matching approach. Trans. Comput. (ToC) **67**(5), 3753–3767 (2019)
8. Baswana, S., Gupta, M., Sen, S.: Fully dynamic maximal matching in $\mathcal{O}(\log n)$ update time (corrected version). J. Comput. (JoC) **47**(3), 617–650 (2018)
9. Behnezhad, S., Lkacki, J., Mirrokni, V.: Fully dynamic matching: Beating 2-approximation in Δ^{ϵ} update time. In: Symposium on Discrete Algorithms (SODA), pp. 2492–2508. SIAM (2020)
10. Bernstein, A., Stein, C.: Fully dynamic matching in bipartite graphs. In: Halldórsson, M.M., Iwama, K., Kobayashi, N., Speckmann, B. (eds.) Automata, Languages, and Programming: 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I, pp. 167–179. Springer Berlin Heidelberg, Berlin, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47672-7_14
11. Bhattacharya, S., Gupta, M., Mohan, D.: Improved algorithm for dynamic b-matching. In: 25th European Symposium on Algorithms (ESA). LIPIcs, vol. 87, pp. 15:1–15:13. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2017)

12. Bienkowski, M., Fuchssteiner, D., Marcinkowski, J., Schmid, S.: Online dynamic b-matching: with applications to reconfigurable datacenter networks. Special Interest Group Comput. Perform. Eval. Commun. (SIGMETRICS) **48**(3), 99–108 (2021)
13. Davis, T.A., Hu, Y.: The university of florida sparse matrix collection. Trans. Math. Softw. (TOMS) **38**(1), 1–25 (2011)
14. Dong, L., Kang, X., Pan, M., Zhao, M., Zhang, F., Yao, H.: B-matching-based optimization model for energy allocation in sea surface monitoring. Energy **192**, 116618 (2020)
15. Drake, D.E., Hougardy, S.: A simple approximation algorithm for the weighted matching problem. Inform. Process. Lett. (IPL) **85**(4), 211–213 (2003)
16. Duan, R., Pettie, S.: Linear-time approximation for maximum weight matching. J. ACM **61**(1) (2014)
17. El-Hayek, A., Hanauer, K., Henzinger, M.: On b-matching and fully-dynamic maximum k-edge coloring. arXiv:2310.01149 (2023)
18. Gabow, H.N.: Data structures for weighted matching and extensions to b-matching and f-factors. Trans. Algorithms (TALG) **14**(3), 1–80 (2018)
19. Hadji, M., Djenane, N., Aoudjit, R., Bouzefrane, S.: A new scalable and energy efficient algorithm for vms reassignment in cloud data centers. In: Future Internet of Things and Cloud Workshops (FiCloudW), pp. 310–314. IEEE (2016)
20. Halappanavar, M., Feo, J., Villa, O., Tumeo, A., Pothen, A.: Approximate weighted matching on emerging manycore and multithreaded architectures. J. on HPC (JoHPC) **26**(4), 413–430 (2012)
21. Henzinger, M., Khan, S., Paul, R., Schulz, C.: Dynamic matching algorithms in practice. In: 28th European Symposium on Algorithms (ESA). LIPIcs, vol. 173, pp. 58:1–58:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2020)
22. Hoefer, M.: Local matching dynamics in social networks. Inform. Comput. **222**, 20–35 (2013). https://doi.org/10.1016/j.ic.2012.10.005
23. Huang, D., Pettie, S.: Approximate generalized matching: f-factors and f-edge covers. Algorithmica **84**(7) (2022)
24. Khan, A., et al.: Efficient approximation algorithms for weighted b-matching. J. on Sci. Comp. (JoSC) **38**(5), S593–S619 (2016)
25. Khan, A., et al.: Designing scalable b-matching algorithms on distributed memory multiprocessors by approximation. In: Supercomputing (SC'16), pp. 773–783. IEEE (2016)
26. Khorasani, F., Gupta, R., Bhuyan, L.N.: Scalable simd-efficient graph processing on gpus. In: Parallel Architectures and Compilation Techniques (PACT), pp. 39–50. IEEE (2015)
27. Kunegis, J.: Konect: the koblenz network collection. In: World Wide Web Conference (WWW), pp. 1343–1350 (2013)
28. Lebedev, D., Mathieu, F., Viennot, L., Gai, A.T., Reynier, J., De Montgolfier, F.: On using matching theory to understand p2p network design. In: International Network Optimization Conference (INOC) (2007)
29. Liu, R., Huang, J., Zhang, Z.: Tracking disclosure change trajectories for financial fraud detection. Prod. Oper. Manag. **32**(2), 584–602 (2023)
30. von Looz, M., Özdayi, M.S., Laue, S., Meyerhenke, H.: Generating massive complex networks with hyperbolic geometry faster in practice. In: High Performance and Embd. Computing (HPEC), pp. 1–6. IEEE (2016)
31. Manne, F., Halappanavar, M.: New effective multithreaded matching algorithms. In: International Parallel and Distributed Processing Symposium (IPDPS), pp. 519–528. IEEE (2014)

32. Mestre, J.: Greedy in approximation algorithms. In: Azar, Y., Erlebach, T. (eds.) Algorithms – ESA 2006, pp. 528–539. Springer Berlin Heidelberg, Berlin, Heidelberg (2006). https://doi.org/10.1007/11841036_48
33. Naim, M., Manne, F.: Scalable b-matching on gpus. In: International Parallel and Distributed Processing Symposium Workshop (IPDPSW), pp. 637–646. IEEE (2018)
34. Perumal, S.S., Lusby, R.M., Larsen, J.: Electric bus planning & scheduling: A review of related problems and methodologies. European J. of Op, Research (2021)
35. Pothen, A., Ferdous, S., Manne, F.: Approximation algorithms in combinatorial scientific computing. Acta Numer **28**, 541–633 (2019)
36. Samineni, B., Ferdous, S., Halappanavar, M., Krishnamoorthy, B.: Approximate bipartite b-matching using multiplicative auction. arXiv:2403.05781 (2024)
37. Ting, H.F., Xiang, X.: Near optimal algorithms for online maximum edge-weighted b-matching and two-sided vertex-weighted b-matching. Theor. Comput. Sci. (TCS) **607**, 247–256 (2015)
38. Wang, Y., Tong, Y., Long, C., Xu, P., Xu, K., Lv, W.: Adaptive dynamic bipartite graph matching: a reinforcement learning approach. In: International Conference on Data Engineering (ICDE), pp. 1478–1489. IEEE (2019)

Harmonizing Vs Polarizing Platform Influence Functions

Hind AlMahmoud[1,2](✉) and Frederik Mallmann-Trenn[1]

[1] King's College London, London, UK
frederik.mallmann-trenn@kcl.ac.uk
[2] Imam Mohammad Ibn Saud Islamic University, Riyadh, Kingdom of Saudi Arabia
hind.almahmoud@kcl.ac.uk

Abstract. We investigate the dynamics of opinion formation on social networking platforms, focusing on how individual opinions, influenced by both social connections and platform algorithms, evolve. We model this process using a differential equation, accounting for both peer influence and the platform's content curation based on user preferences. Our primary aim is to analyze how these factors contribute to opinion polarization and identify potential strategies for its mitigation. We explore the conditions under which opinions converge to a consensus or remain polarized, emphasizing the role of the platform's influence function. Our findings in two-agent, complete graphs, and stochastic block model provide insights into the impact of social media algorithms on public discourse and offer a framework for understanding how polarization can be avoided.

Keywords: Persistent disagreement · Equilibrium · Consensus · Polarization · Social learning · Opinion dynamics

1 Introduction

Research shows that platforms and media can significantly influence people's opinions. For instance, 14% of Americans reported changing their views due to something they saw on social media, with younger users being more affected [8], which highlights the influential role of media. Furthermore, social networks play a critical role in shaping political decisions and disseminating opinions, as demonstrated during events like the 2008 U.S. presidential election [11]. As these platforms have become integral to our daily interactions, the content they present—often tailored to individual user preferences—can substantially shape our views. This customization tends to favor content that aligns with users' existing beliefs, which can potentially lead to increased polarization. In this paper, we examine the impact of social networking platforms, such as Instagram and Facebook, on the formation of opinions and beliefs. Our research aims to deepen the understanding of this polarization process and explore potential avenues for its mitigation. We approach this by modeling a network of users, where each

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 152–165, 2025.
https://doi.org/10.1007/978-3-031-82439-5_13

individual at any given time t has an opinion $x_i(t)$ within a range of $[-1, 1]$. Each opinion evolves based on an equation that considers both the influence of others and the platform's algorithmic impact, represented as:

$$\dot{x_i}(t) = \sum_{j \neq i}^{n} a_{ij}(x_j(t) - x_i(t)) + b \cdot (f(x_i(t)) - x_i(t)), \tag{1}$$

where $f(\cdot)$ is the platform's influence function, n is the number of users and b is a positive constant. We denote the vector of functions x as an *opinion dynamic* if it starts at a feasible point (in $[-1, 1]^n$) and then evolves as described by the ODE (1). This model was introduced by [12], who studied $f(\cdot)$ for the sign function. The influence parameter $a_{ij} \geq 0$ can take any value greater than zero. For simplicity, we assume that $a_{ij} = a_{ji}$, i.e., the matrix $A = (a_{ij})$ is symmetric. Our paper builds on existing models to analyze how the platform function affects opinion dynamics, particularly focusing on the system's convergence to stable states and the potential of the platform function to either exacerbate or mitigate polarization. There are two types of final states persistent disagreement states in which the agents will never reach consensus, with their opinion unchanging, and consensus, where agents align on a common opinion, meaning their final values share the same sign. Strong consensus is a more specific case where all agents not only agree in general but hold exactly the same opinion. Our contribution is under mild assumptions on $f(\cdot)$, such as continuity, symmetry, sign preservation, and boundedness, (see Sect. 3.1) we give sufficient and necessary conditions for persistent disagreement states. See Theorem 2, 3, and 5. In addition, we characterize all the states where strong consensus is reached in Theorem 1 and 4. Also, we generalise our results to stochastic block model of two blocks as seen in Sect. 6. Finally, we analyse a set of harmonizing and polarizing functions in Sect. 7, where we show for a function $f(\cdot)$ that violate sign preservation, that the system always converges to consensus see Theorem 7. The structure of this paper is as follows: Sect. 3 introduces the model, detailing the classification of platform functions. The results are divided into four parts and presented in Sects. 4,5, 6, and 7. In Sects. 4 and5, we examine the outcomes for general platform functions in a two-agent system and complete graphs. In Sect. 6 we generalise our results to stochastic block model of two blocks, and Sect. 7 is dedicated to the results for specific platform functions. The paper concludes with Sect. 8, where we summarize our findings and discuss potential future work.

2 Related Literature

Over the past fifty years, mathematical models such as the DeGroot [14] and Friedkin-Johnsen (FJ) [18] models have been central to analyzing social learning and opinion dynamics. The DeGroot model illustrates iterative opinion updates based on neighbor influences until convergence. The FJ model expands this by including external information and individual parameters like confidence and susceptibility, allowing for more complex opinion dynamics.

Several social theories such as herd behavior, informational cascade, and homophily have been explored to understand opinion dynamics [6,9,13]. Dandekar et al. [13] extends DeGroot's model to include biased assimilation, revealing that polarization in homophilous networks arises only under strong bias, indicating homophily alone doesn't sufficiently polarize society.

Recent research has increasingly focused on how media influences opinion dynamics and polarization. For instance [26] studies opinion dynamics under algorithmic personalization, exploring the effects of various filtering algorithms. A research more closely aligned with our paper is [12], the continuous model in this paper bears resemblance to that of [12] and draws inspiration from it. Our results deviate from theirs in that we offer findings for broad categories of platform functions that are harmonizing and polarizing. We also provide specific examples of functions for which we have pinpointed all potential steady states.

Also, two recent research includes the media in opinion dynamics in [3] they investigate how social media recommendations affect opinion dynamics, agents are assigned discrete opinions. Their results show that strong social media influence can lead to polarization, while a high homophily can reduce the impact of social media, which reduces consensus probability. In contrast,[20] focuses on continuous opinions, where agents hold opinions evolve in real numbers.

While there is a continuing discussion on whether the Internet and social media amplify or mitigate polarization as shown in [25]. Researchers have raised concerns regarding the potential role of social media in driving political polarization such as [1,10,15,16,19,21,22,28]. A study by [1] supports the idea that social media platforms, particularly Facebook, may contribute to polarization. Additionally, [21] shows that internet access may also contribute to an increase in polarization. Also, [10] explores political polarization in U.S. adults using eight different measures, while [22] conducted a field experiment on Facebook, offering news subscriptions from various political spectra to assess impacts on user attitudes and news consumption. Results indicate that while ideologically aligned news reduces negativity towards opposing views, Facebook's algorithm may limit exposure to opposing perspectives, potentially increasing polarization. Moreover, [15,16] use Twitter data to study the dynamics of political and climate change polarization, highlighting the role of social media influencers and the significance of understanding these effects. The research in [19,24,28,29], and [23] explore polarization on social media, in [19] focuses on how Facebook's algorithms affect political polarization, while [28] develops a model to analyze affective polarization through concepts like echo chambers and filter bubbles. Also, [7] models the dynamics of radicalization and echo chamber formation, demonstrating how user engagement on social media can lead to more extreme views. The results emphasize the reasons that may contribute to the existence of echo chambers and polarization in social media networks. Additionally, [4] investigates how social media structures and fake news contribute to misinformation and polarization, noting that significant effects can occur even if only 15% of users believe in fake news.

The papers [5,22], and[17] all support the idea that social media platforms can increase political polarization. In addition, in [2] the authors studied how polarized opinions spread through social networks by using differential equations to describe the dynamics of opinion change over time. In contrast to the aforementioned research, the analysis in [10] reveals an opposite result which is that polarization has grown the most in demographic groups that use the internet and social media the least which means the Internet and social media may lead to decreased polarization.

Although the research community has extensively studied polarization. However, there is still much to learn and many unanswered questions to explore, and ongoing research in this field is essential for developing a deeper understanding of polarization and how to mitigate its negative effects.

3 Model and Notation

For $k \in \mathbb{N}$, let $[k] = \{1, 2, \ldots k\}$. Let V be the set of all agents/vertices and $n = |V|$. Throughout the paper we assume all agents are connected, i.e., in a clique except in Sect. 6. The influence matrix of this graph is represented by a matrix A. The (i, j)-th entry $a_{ij} \geq 0$ in A denotes the influence of agent j's on agent i. In this paper, we assume that the matrix A is symmetric, meaning $a_{ij} = a_{ji}$. We use the model proposed by [12], where agents update their opinion based on their neighbours' opinions and the platform's influence.In this model, given a set of n agents, the opinion of each agent $i \in [n]$ at time $t \geq 0$ is denoted by $x_i(t) \in [-1, 1]$. Here, the domain of $x_i(t)$ signifies that the opinions can range from strongly negative (-1) to strongly positive (+1), encapsulating all possible opinion states. We assume that we have n agents, and every agent is connected to all the other agents. In this paper we consider the system of two agents, which can also be interpreted as two cliques connected (where each agent inside a clique/community has the same initial value), complete graphs, and stochastic block model of two blocks. To ensure that the opinions do not exceed the bounds of $[-1, 1]$, the platform functions are designed to return values within this range only. Each agent's opinion is influenced by two forces, by their connections and the platform function. Let $b > 0$ present the strength of the platform's influence. The opinions are posited to evolve according to (1),and in the case of the two agent system. For $i, j \in \{1, 2\}$ with $j \neq i$, the differential equation is given by

$$\dot{x_i}(t) = a_{ij}(x_j(t) - x_i(t)) + b \cdot (f(x_i) - x_i(t)) \tag{2}$$

where $\dot{x_i}(t)$ indicates the rate of change of x_i over time, and $f : [-1, 1] \to [-1, 1]$ is the function employed by the platform to curate content for the user. The opinion dynamics presented in (1) suggests that shifts in an agent's stance are influenced in two ways: the agent's social interactions and the platform's curated content. The first term embodies the continuous social learning akin to the naive (DeGroot-like) model [14], while the second term encapsulates the platform's force, determined by the strength of the platform's influence b, and the deviation between the agent's current opinion and the platform's targeted content. We now give the key definitions we require for our results.

Definition 1 (Equilibrium State or Steady State). *We say an opinion dynamic x is in equilibrium at time t^* if for all $t \geq t^*$ and all agents i, it holds that $\dot{x}_i(t) = 0$. A vector or state $(y_1, y_2, \ldots, y_n)$ is an equilibrium, if there is some opinion dynamic that attains its equilibrium in this state.*

Note that our definition, as well as the definition of opinion dynamic, imply that a point $(y_1, y_2, \ldots, y_n)$ is an equilibrium or steady state if an opinion dynamic initiated at this point—specifically, $x_1(0) = y_1$, $x_2(0) = y_2$, $\ldots$, $x_n(0) = y_n$ remains constant over time. We divide these equilibria into two categories:

Definition 2 ((Strong) Consensus). *The system is in consensus at time t if it is in a equilibrium and for all $i \in V$, $x_i(t) > 0$, for all $i \in V$, $x_i(t) < 0$, or for all $i \in V$, $x_i(t) = 0$. Again, we call a point a state of consensus if starting at an opinion dynamic in this state, results in consensus for all $t \geq 0$. Furthermore, we say the system is in* strong consensus *at time t if it is in consensus at time t and $\forall i, j \in [n], x_i(t) = x_j(t)$, indicating all agents in the graph have the same value.*

Observe that, $(0, 0, \ldots, 0)$ is always a strong consensus state, due to $f(0) = 0$ (which is a consequence of the assumptions we make regarding Platform Influence Functions Properties).

Definition 3 (Persistent Disagreement). *The system is in persistent disagreement (PD) at time t if it is in a equilibrium and there exists $i, j \in V$ such that $x_i(t) > 0$ and $x_j(t) < 0$. We will sometimes refer to such states as being polarized. We say a point is in persistent disagreement if starting an opinion dynamic at it results in a state of persistent disagreement for all $t \geq 0$.*

As we will see later, the existence of consensus states and persistent disagreement states are determined by the platform influence function. We explore a spectrum of platform influence functions, which we classify into two distinct categories based on their dynamic impact on the system:

Definition 4 (Harmonizing Platform Influence Functions-HPIF). *We say a platform influence function $f(\cdot)$ is harmonizing, if applying this function over time leads to a state where there is no persistent disagreement state.*

Definition 5 (Polarizing Platform Influence Functions-PPIF). *We say a platform influence function $f(\cdot)$ is polarizing if applying this function over time leads to at least one persistent disagreement state.*

3.1 Platform Influence Functions Properties

We will assume that all platform influence functions adhere to the following properties:

1. **Continuity:** Each function is continuous and Lipschitz continuous, ensuring a smooth transition of influence across the range of values.

2. **Symmetry:** They exhibit symmetric behavior about the origin, governed by the property $f(-x) = -f(x)$.
3. **Sign Preservation:** The functions return a positive value for positive inputs, and a negative value for negative inputs.
4. **Boundedness:** To ensure that opinions remain within the range of possible bias $[-1, 1]$, we require that $f(\cdot) \in [-1, 1]$.

As we denote a function fulfilling properties 1 to 4 a *feasible function*. Examples of this family include, but are not limited to: the sgn_ε function, and linear functions $f(x) = \alpha x$ for $\alpha < 1$ (see (7.2)).

3.2 Polarization Planes in Two Agent System

It is well-known [27] that for continuous f a solution to the differential equation exists, and according to Theorem 3 in [12] there exists an equilibrium that agents converge to it in the limit if the influence matrix A is symmetric. To answer the question of whether the equilibrium is a state of consensus or persistent disagreement in the case of two agent system, we introduce the following concept of polarization planes. We define a polarization plane to be a map from $[-1, 1] \times [-1, 1] \times (-\infty, \infty) \to (-\infty, \infty)$, let G and H be two polarization planes to be *polarization planes*, which we will use to delineate critical thresholds in the opinion dynamics model. As we will see later, they define equilibrium states where the gradient of the system's potential is zero. For simplicity in explaining the planes we assume that $a_{ij} = a_{ji} = 1$. Based on the ordinary differential equation presented in (2), The system will be in an equilibrium if and only if for all $i, j \in [2]$ with $j \neq i$, $f(x_i(t)) = \frac{x_i(t)-x_j(t)}{b} + x_i(t)$. We define the polarization planes, to include all points (x, y) that satisfy $f(x) = \frac{x-y}{b} + x$ and $f(y) = \frac{y-x}{b} + y$. These points can be represented as three-dimensional planes, namely $(x, y, \frac{x-y}{b}+x)$ and $(x, y, \frac{y-x}{b}+y)$. See Fig. 1 for an illustration. We define, $G(x, y, b) = \frac{x-y}{b}+x$ and $H(x, y, b) = \frac{y-x}{b}+y$. Similarly, we can define the plane of values to be $F_1 = \{(x, y, f(x)) \mid x, y \in [-1, 1]\}$ and $F_2 = \{(x, y, f(y)) \mid x, y \in [-1, 1]\}$. . The full paper can be found on Arxiv which provides some key insights into the polarization planes.

Persistent Disagreement Results. Using the polarization planes and platform function planes we can summarize our results for persistent disagreement into the following three cases. Due to the symmetry mentioned before, we will only focus on the octant where $x > 0, y < 0$, and $f(x) \geq 0$.

- The plane F_1 intersects the G plane at least at one point, i.e., $F_1 \cap G \neq \emptyset$. Due to symmetry, F_2 must also intersect the H plane. Then, there exists at least one persistent disagreement state. See Theorem 2.
- The entire plane F_1 is located below the G plane and, due to its symmetry, F_2 is above H. In this case, there is no persistent disagreement state see Theorem 3.

– The entire plane F_1 is located above the G, i.e., for all $x > 0$ and $y < 0$, $f(x) \geq \frac{x-y}{b} + x$. Due to symmetry, F_2 must be completely below the H plane. This cannot occur, because there is no F_1 completely above G since the platform function is, by definition, bounded by 1 while the plane G exceeds 1.

For item 1, we call this function the polarizing function as we will elaborate in Sect. 7.1. For item 2, we call this function the harmonizing function as we will elaborate in Sect. 7.2.

3.3 Polarization Conditions in Complete Graphs

The concept of polarization planes can be extended for complete graphs with n vertices which we denote as K_n. Since we have complete graph, then each agent i has a fixed number of connections $n-1$ connections. Therefore, the weight of each connection $\frac{a}{n-1}$, i.e., $a_{ij} = \frac{a}{n-1}$ is the influence of agent j on agent i. Hence, the opinions of agent i evolve according to the equation

$$\dot{x}_i(t) = \frac{a}{n-1} \sum_{j \in N(i)} (x_j(t) - x_i(t)) + b\left(f\left(x_i(t)\right) - x_i(t)\right)$$

where n is the total number of agents in the network, $a > 0$ is the influence parameter. In the equilibrium where $\dot{x}_i(t) = 0$ we have $f(x_i) = \frac{(a+b)x_i(t) - a\bar{x}_i(t)}{b}$. where $\bar{x}_i(t)$ is the mean of all the neighboring nodes of node i in the complete graph (excluding i), it can be written as: $\bar{x}_i(t) = \frac{1}{n-1}\sum_{j \in N(i)} x_j(t)$. To be consistent with the two agent system lets call this function G such that:

$$G(x_i(t), \bar{x}_i(t), a, b) = \frac{(a+b)x_i(t) - a\bar{x}_i(t)}{b}$$

If there exists a time $t^* \geq 0$ such that $G(x_i(t^*), \bar{x}_i(t^*), a, b) = f(x_i(t^*))$, for every agent i in the state vector $\mathbf{x} = (x_1(t), x_2(t), \ldots, x_n(t))$ of all agents in the graph, then all agents in K_n have reached equilibrium. We can define the plane of values as $F_i = \{(x_i(t), \bar{x}_i(t), f(x_i(t))) \mid x_i(t), \bar{x}_i(t) \in [-1, 1]\}$, More details and some key insights into the polarization planes in complete graphs can be found in the full paper.

4 Results for General Platform Functions for Two-Agent System

In this section we present our results of characterizing necessary and sufficient conditions for strong consensus and persistent disagreement states in the case of the two-agent system.

Theorem 1 (Sufficient and Necessary Conditions for Strong Consensus). *Fix arbitrary $b > 0$ and $a > 0$, and consider an opinion dynamic x with two agents i and j. Assume $f(\cdot)$ fulfills the Platform Influence Functions Properties. Then, for any strong consensus state (y, y) it is sufficient and necessary that $f(y) = y$.*

Observe that Theorem implies that there is at least one consensus state in the quadrant $(+,+)$ and one in the quadrant $(-,-)$. The proof of this Theorem can be seen in full paper.

Theorem 2 (Sufficient Condition for PD). *Fix arbitrary $b > 0$ and $a > 0$, and consider an opinion dynamic x with two agents i and j. Assume $f(\cdot)$ fulfills the Platform Influence Functions Properties. If there exist $z \in (0,1]$ and $y \in [-1,0)$ such that $f(z) \geq G(z,y,a,b)$ and $f(y) \leq H(z,y,a,b)$, then there exists at least one persistent disagreement state.*

The proof of this Theorem can be seen in full paper.

Theorem 3 (Necessary Condition for PD). *Fix an arbitrary $b > 0$ and $a > 0$. Consider an opinion dynamic x with two agents i and j. Assume $f(\cdot)$ fulfills the Platform Influence Functions Properties. If a persistent disagreement state exists at $x_i(t)$ and $x_j(t)$, then there must exist $z \in (0,1]$ and $y \in [-1,0)$ such that $f(z) \geq G(z,y,a,b)$ and $f(y) \leq H(z,y,a,b)$.*

The proof of this Theorem can be seen in full paper.

5 Results for General Platform Functions for Complete Graph of N Agents

In this section we present the necessary and sufficient conditions for strong consensus and persistent disagreement states in complete graphs consisting of n agents.

Theorem 4 (Strong Consensus). *Fix arbitrary $b > 0$ and $a > 0$, and consider a complete graph with n vertices, K_n. Assume $f(\cdot)$ fulfills the Platform Influence Functions Properties. Then, for any strong consensus state $(y, y, \ldots, y) \in \mathbb{R}^n$ it is sufficient and necessary that $f(y) = y$ where $y \in [-1,1]$.*

Theorem 5 (Necessary Condition for PD). *Fix arbitrary $b > 0$ and $a > 0$, and consider a complete graph K_n with n agents, where each agent i has an opinion $x_i(t)$ at time t. Define:*

- *$X^+(t) = \{x_i(t) \mid x_i(t) \geq 0\}$ to be the set of states of agents with non-negative opinions.*
- *$X^-(t) = \{x_j(t) \mid x_j(t) \leq 0\}$ to be the set of states of agents with non-positive opinions.*

Assume $f(\cdot)$ fulfills the Platform Influence Functions Properties, and where $\bar{x}_i(t)$ is the mean of all the neighboring nodes of node i . If a persistent disagreement state exists then there must exist at least one agent in $X^+(t)$ and $X^-(t)$, such that

- *For all $x_i(t) \in X^+(t)$, $f(x_i(t)) \geq G(x_i(t), \bar{x}_i(t), a, b)$*
- *For all $x_j(t) \in X^-(t)$, $f(x_j(t)) \leq G(x_j(t), \bar{x}_i(t), a, b)$*

The proof of these two Theorems can be seen in full paper.

6 Results for the Stochastic Block Model with Two Blocks

In this section, we extend the results from the two-agent system to the stochastic block model (SBM) with two blocks. The equilibrium of the two-agent system can be approximated in large, densely connected networks. Inspired by Theorem 2 in [12], we generalize the result for functions that adhere to platform properties. The main equation and notations are inspired by Theorem 2 in [12].We consider two sets of agents, left and right, with intra- and inter-group connection probabilities. The initial opinions of the agents are random variables assigned from predefined sets for each block. Using mean-field approximation, we normalize the influence matrix A to a scalar a. The influence of a connected agent j on agent i is given by $\frac{a}{d_i}$, where d_i is the number of connections of i. The opinion of agent i evolves as:

$$\dot{x}_{i,n}(t) = \frac{a}{|N(i)|} \sum_{j \in N(i)} \left(x_{j,n}(t) - x_{i,n}(t)\right) + b\left(f(x_{i,n}(t)) - x_{i,n}(t)\right),$$

where $N(i)$ denotes the connections of agent i. The main result is that in dense networks, the opinions of agents in the left and right blocks converge to the equilibrium of a two-agent system, assuming certain initial conditions. We define A_n as an adjacency matrix, and the set of all possible adjacency matrices for a network of n left and right agents is denoted by $\boldsymbol{A}_n$. The SBM generates a probability space $\left(\boldsymbol{A}_n, 2^{\boldsymbol{A}_n}, \mathbb{P}\right)$, where $\mathbb{P}$ is the probability that a matrix A_n is realized. The initial opinions of left (right) agents are random variables on $X_L \subset \mathbb{R}_-$ ($X_R \subset \mathbb{R}_+$), and the distribution is captured by the measure μ_0. Given a realized adjacency matrix A, the opinion of each agent evolves based on connections and platform influence. The main theorem which can be seen in full paper shows that, under sufficient network density and specific initial conditions, agents' opinions approximate those in a two-agent system. In dense networks, where $p(n)$ and $q(n)$ satisfy $\omega\left(\frac{\ln(n)}{n}\right)$, expected cross-block connections are $q(n)n$, and total connections are $p(n)n + q(n)n$. Applying Lemma 4 from [12], each agent has approximately $q(n)n$ opposite-type connections and $p(n)n + q(n)n$ total connections. For large t, the opinions of all agents in both blocks approach the equilibrium of a representative agent in a two-agent system. The opinion of each agent is primarily influenced by the opposite-type representative agent at a rate $a\beta$, where $\beta = \frac{q(n)n}{p(n)n+q(n)n}$. Therefore, the equilibrium states of the SBM with two blocks approximate those of the two-agent system for large n and t. The details of how we proved that can be seen in full paper.

7 Results for Specific Platform Functions

In this section, we study explicit platform signaling functions and consider both polarizing and harmonizing functions.

7.1 Examples of Polarizing Platform Influence Functions (PPIF)

One example of a polarizing platform influence function is the sign function as described below.

Sign Function. This function was used by [12] as their platform function. This platform function pushes each agent's opinion towards content that aligns with their current view. It is a polarizing function as it intersects with the polarization planes as shown in the Figs. 1, 2 and 3 for the $(+,-)$ quadrant. The purple plane represents G, the green plane represents H, the red plane represents $(x_i, x_j, f(x_i))$, and the yellow plane represents$(x_i, x_j, f(x_j))$ with $\varepsilon = 0.05$, where ε is a threshold. In [12] the authors present a continuous approximation of the sign function for a very small ε>0. The continuous approximation of the sign function based on a small positive value ε $sgn_\varepsilon(x_i)$ is as follows: for $x_i < -\varepsilon$, the function returns -1. For $-\varepsilon \leq x_i \leq \varepsilon$, it returns x_i/ε, and for $x_i > \varepsilon$, it returns 1. In a two agent opinion dynamic with some network influence $b > 0$, and with the agent's influence on the other agent given by $a_{12} = a_{21} = a$ for some $a > 0$. There are five steady states: They are $(1,1)$, $(-1,-1), (0,0), (\frac{b}{2a+b}, \frac{-b}{2a+b})$ and $(\frac{-b}{2a+b}, \frac{b}{2a+b})$, as shown in [12]. We generalize this prove to n agents in the full paper, but it is useful to first understand the case of two agents. We visualize the aforementioned equilibria in Fig. 4, the green points show the consensus points, and the red points show the PD points. Since it is a polarization function, then F_1 and G intersect in the point $(\frac{b}{2a+b}, \frac{-b}{2a+b})$. For full proof see full paper.

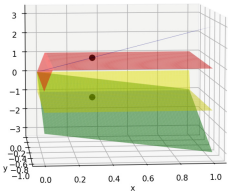


Fig. 1. A depiction of the sign function along with the intersection points and polarization planes: G (purple), F_1 (red), F_2 (yellow) and H (green).

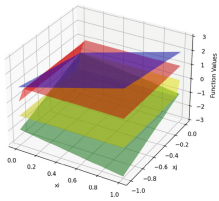


Fig. 2. A depiction of the sign function and polarization planes: G (purple), F_1 (red), F_2 (yellow) and H (green).

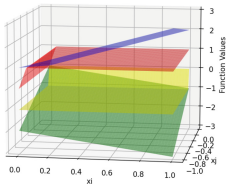


Fig. 3. A depiction of the sign function and polarization planes: G (purple), F_1 (red), F_2 (yellow) and H (green) from a different viewing angle.

7.2 Examples of Harmonizing Platform Influence Functions(HPIF)

This section presents two examples of harmonizing platform influence functions.

The Linear Harmonizing Function. Fix $0 < \alpha < 1$, the first platform function is $f(x_i) = \alpha x_i$ Since $\alpha < 1$, one can show (Theorem 6) that the system will always converge to the equilibrium $(0, 0, \ldots, 0)$. From the point of view of a platform such a function still keeps users on the platform, because the content they see, governed by $f(\cdot)$, is close to their own. However, it gradually pushes users towards a state of consensus. It is a harmonizing function, therefore, it does not intersect with the polarization planes as shown in Fig. 5. To show that this function never intersects with the plane G. Lets consider $x_i \in (0, 1]$ and $x_j \in [-1, 0)$ to intersects with G, the following equation has to hold true. $\alpha x_i = \frac{a(x_i - x_j)}{b} + x_i$ Subtract x_i from both sides: $(\alpha - 1)x_i = \frac{a(x_i - x_j)}{b} + x_i$ This can never be true as the left-hand side of the equation is always negative and right-hand side of the equation is always positive.

Theorem 6. *Consider a clique of n agents and the following platform signal function. There is only a stable solution which is the state where all agents have value 0, i.e., for all $i \in [n]$, $x_i = 0$. $f(x_i) = \alpha x_i$*

The proof of this Theorem can be seen in full paper.

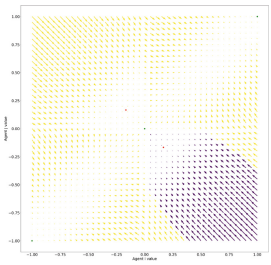


Fig. 4. Vector field plot of Sign Function in two agents system where $b = 0.4$ and $a = 1$. Green points indicate consensus while red ones represent persistent disagreement.

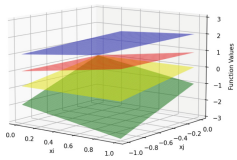


Fig. 5. The Linear Harmonizing Function and Polarization Planes with $\alpha = 0.9$.

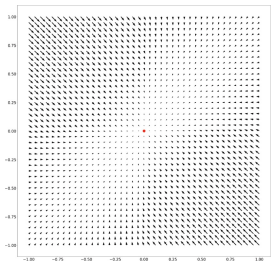


Fig. 6. Vector field plot of Anti Sign Function in two agents system showing all points converging to the origin.

Anti Sign Function. This platform function presents the user with content that opposes their initial opinion. This can be beneficial in promoting diverse perspectives and reducing polarization. For example, if a user expresses agreement with a particular topic or viewpoint, the platform will show them content that presents an opposing viewpoint or opinion. This function can encourage users to consider alternative perspectives and make more informed decisions. In a two-agent system and some network influence $b > 0$,and peer influence $a > 0$ then according to Eq. 2, there is one stable state as shown in Fig. 6.

Theorem 7. *Consider a clique of n agents and the $antiSgn_\varepsilon(x_i)$ as the platform function. There is only a stable solution which is* $(0, 0, \ldots, 0)$.

Proof. The proof is similar to Theorem 6 where there is only one steady state which is the origin.

The anti-sign function is defined as follows:

$$\text{antiSgn}_\varepsilon(x_i) = \begin{cases} 1 & \text{if } x_i < -\varepsilon, \\ -x_i/\varepsilon & \text{if } -\varepsilon \leq x_i \leq \varepsilon, \\ -1 & \text{if } x_i > \varepsilon. \end{cases}$$

Please note that the anti-sign function violates the sign preservation property. Moreover, realistically speaking, a platform would avoid showing a user constantly content that is far from its opinion as it will likely cause the user to feel alienated and disengage with the platform.

8 Conclusion and Future Work

In this paper, we show how individual viewpoints shaped by social interactions are influenced by the platform by studying the platform influence function. We study when strong consensus and persistent disagreement is possible by giving the necessary and sufficient conditions for both. To achieve this we study a concept we coin polarisation planes that reveal interesting geometric properties. Moreover, we generalise these results to clique of n agents, also we proved that our results can be generalized to stochastic block model of two blocks. Studying the regions of attraction - meaning to which equilibrium state the system converges depending on the initial state - remains an interesting open question. We also provide some functions that avoid persistent disagreement and it would be interesting to see if such functions help in the real world to combat polarization in social networks.

Acknowledgements. We are very grateful to Laura Vargas Koch and Víctor Verdugo for their help in writing the paper. Part of the research was supported by the EPSRC grant EP/W005573/1.

References

1. Allcott, H., Braghieri, L., Eichmeyer, S., Gentzkow, M.: The welfare effects of social media. AM. Econ. Rev. **110**(3), 629–676 (2020)
2. Amelkin, V., Bullo, F., Singh, A.K.: Polar opinion dynamics in social networks. IEEE Trans. Autom. Control **62**(11), 5650–5665 (2017)
3. Auletta, V., Coppola, A., Ferraioli, D.: On the impact of social media recommendations on consensus of discrete opinions (2023)
4. Azzimonti, M., Fernandes, M.: Social Media Networks, Fake News, and Polarization, Technical report, National Bureau of Economic Research (2018)

5. Bakshy, E., Messing, S., Adamic, L.A.: Exposure to ideologically diverse news and opinion on Facebook. Science **348**(6239), 1130–1132 (2015)
6. Banerjee, A.V.: A simple model of herd behavior. Q. J. Econ. **107**(3), 797–817 (1992)
7. Baumann, F., Lorenz-Spreen, P., Sokolov, I.M., Starnini, M.: Modeling echo chambers and polarization dynamics in social networks. Phys. Rev. Lett. **124**(4), 048301 (2020)
8. Bialik, K.: 14% of Americans have changed their mind about an issue because of something they saw on social media (2018). https://www.pewresearch.org/short-reads/2018/08/15/14-of-americans-have-changed-their-mind-about-an-issue-because-of-something-they-saw-on-social-media/
9. Bikhchandani, S., Hirshleifer, D., Welch, I.: A theory of fads, fashion, custom, and cultural change as informational cascades. J. Polit. Econ. **100**(5), 992–1026 (1992)
10. Boxell, L., Gentzkow, M., Shapiro, J.M.: Greater internet use is not associated with faster growth in political polarization among us demographic groups. Proc. Natl. Acad. Sci. **114**(40), 10612–10617 (2017)
11. Burbach, L., Halbach, P., Ziefle, M., Calero Valdez, A.: Opinion formation on the internet: the influence of personality, network structure, and content on sharing messages online. Front. Artif. Intell. **3**, 45 (2020)
12. Candogan, O., Immorlica, N., Light, B., Anunrojwong, J.: Social learning under platform influence: consensus and persistent disagreement. arXiv preprint arXiv:2202.12453 (2022)
13. Dandekar, P., Goel, A., Lee, D.T.: Biased assimilation, homophily, and the dynamics of polarization. Proc. Natl. Acad. Sci. **110**(15), 5791–5796 (2013)
14. DeGroot, M.H.: Reaching a consensus. J. Am. Stat. Assoc. **69**(345), 118–121 (1974)
15. Falkenberg, M., et al.: Growing polarization around climate change on social media. Nat. Clim. Chang. **12**(12), 1114–1121 (2022)
16. Flamino, J., et al.: Political polarization of news media and influencers on Twitter in the 2016 and 2020 US presidential elections. Nat. Hum. Behav. **7**, 1–13 (2023)
17. Flaxman, S., Goel, S., Rao, J.M.: Filter bubbles, echo chambers, and online news consumption. Pub. Opin. Q. **80**(S1), 298–320 (2016)
18. Friedkin, N.E., Johnsen, E.C.: Social influence and opinions. J. Math. Sociol. **15**(3–4), 193–206 (1990)
19. Garcia, D.: Influence of Facebook algorithms on political polarization tested (2023)
20. Hu, H., Chen, W., Hu, Y.: Opinion dynamics in social networks under the influence of mass media. Appl. Math. Comput. **482**, 128976 (2024). https://doi.org/10.1016/j.amc.2024.128976
21. Lelkes, Y., Sood, G., Iyengar, S.: The hostile audience: the effect of access to broadband internet on partisan affect. Am. J. Polit. Sci. **61**(1), 5–20 (2017)
22. Levy, R.: Social media, news consumption, and polarization: evidence from a field experiment. Am. Econ. Rev. **111**(3), 831–870 (2021)
23. Pansanella, V., Rossetti, G., Milli, L.: From mean-field to complex topologies: network effects on the algorithmic bias model. In: Complex Networks and Their Applications X: Volume 2, Proceedings of the Tenth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2021 10, pp. 329–340. Springer (2022)
24. Pansanella, V., Rossetti, G., Milli, L.: Modeling algorithmic bias: simplicial complexes and evolving network topologies. Appl. Netw. Sci. **7**(1), 57 (2022)
25. Pappalardo, L., et al.: A survey on the impact of AI-based recommenders on human behaviours: methodologies, outcomes and future directions. arXiv preprint arXiv:2407.01630 (2024)

26. Perra, N., Rocha, L.E.: Modelling opinion dynamics in the age of algorithmic personalisation. Sci. Rep. **9**(1), 1–11 (2019)
27. Pouso, R.L.: Peano's existence theorem revisited. arXiv preprint arXiv:1202.1152 (2012)
28. Törnberg, P., Andersson, C., Lindgren, K., Banisch, S.: Modeling the emergence of affective polarization in the social media society. PLoS ONE **16**(10), e0258259 (2021)
29. Valensise, C.M., Cinelli, M., Quattrociocchi, W.: The drivers of online polarization: fitting models to data. Inf. Sci. **642**, 119152 (2023)

Bots of a Feather: Mixing Biases in LLMs' Opinion Dynamics

Erica Cau[1,2](✉), Andrea Failla[1,2], and Giulio Rossetti[2]

[1] Department of Computer Science, University of Pisa, 56127 Pisa, Italy
{erica.cau,andrea.failla}@phd.unipi.it
[2] ISTI-CNR, National Research Council, 56124 Pisa, Italy
giulio.rossetti@isti.cnr.it

Abstract. The rapid integration of Large Language Models (LLMs) into everyday applications raises critical questions about their group interactions, consensus formation, and potential to mimic human-like behavior. Although initial research has explored the evolution of opinions within LLM populations, these efforts often rely on simplistic network assumptions, such as uniform connections among agents, thereby overlooking the influence of more realistic network topologies. This paper introduces a framework for examining opinion dynamics among LLM agents within various network structures. We perform several multi-model simulations on network topologies with known locally assortative/disassortative mixing patterns. We find that convergence is quicker in mostly-disassortative networks compared to networks with no mixing biases. However, the joint effect of assortative and disassortative patterns leads to slower/no convergence.

Keywords: Large Language Models · Opinion Dynamics · Homophily · Social Networks

1 Introduction

Human societies are gradually integrating Artificial Intelligence (AI) models in their daily lives, to solve a variety of both trivial and complex tasks. Large Language Models (LLMs) are instances of these models that are able to simulate coherent human-like conversations [3], mimic behaviors of specific personae [20], and even achieve remarkable accuracy in complex natural language processing tasks such as translation [3], sentiment analysis [1], and stance detection [23]. As the presence of these models becomes more widespread, it is essential to understand how they interact in large groups, reach consensus and/or disagreement, and whether they can replicate human-like behaviors without any specific training but as an emergent ability [41]. In this direction, recent work in psychology suggests that LLMs might have developed a *Theory of Mind* [4,11,21], i.e., the ability to impute mental states to themselves and others. If they do have a Theory of Mind, then LLMs are able to manifest emotions, beliefs and

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 166–176, 2025.
https://doi.org/10.1007/978-3-031-82439-5_14

desires in a coherent way; most importantly, they are able to understand/predict these mental states when interacting with others. Consequently, LLMs' beliefs could be studied/modeled in the same ways we currently study/model human ones [11,36]. Some works have already investigated how the opinions of LLM populations evolve by leveraging statistical physics and network analysis tools [6,7,39]. Nevertheless, these studies often assume mean-field topologies (i.e., settings where each agent is connected with/affected by all other agents), and the effects of network topology remain largely unexplored. In this work, we address this research gap by introducing a general framework for opinion dynamics with LLM agents, describing an algorithm for LLM debates, and performing various simulations on different network configurations. We aim to understand whether locally assortative/disassortative initial conditions affect the evolution of LLMs' opinions.

The rest of this work is organized as follows. In Sect. 2, we provide an overview of the main topics related to our study; Sect. 3 formalizes a framework for studying LLMs' Opinion Dynamics; Sect. 4 introduces the experimental campaign and discusses the results. Finally, in Sect. 5, we conclude the work and suggest future research directions.

2 Related Work

In this section, we provide the reader with background on the main themes related to our study. First, we outline classical opinion dynamic models; then, we move to more complex LLM-based simulations.

Opinion Dynamics. Opinion dynamics (OD) can be understood as a process simulating how individuals' opinions change over time due to empirically observed social phenomena [5,37]. OD simulations typically involve a set of connected agents (the *population*) holding opinions encoded as real numbers. Social phenomena are implemented as rules that govern how a population's individual opinions are updated over time, leveraging statistical physics tools, such as graph and probability theories. For instance, an individual's opinion could be updated by randomly copying a neighbor [19] (which captures the tendency to mimic others) by averaging the opinions in its neighborhood [13] (which captures homophilic tendencies [25]), by computing the most frequent opinion therein [22] (which captures peer pressure and mutual awareness effects [10]). The goal of OD is generally to understand whether the interplay of the considered social mechanisms eventually leads to one of the following mutually exclusive scenarios: (i) consensus, when all agents hold the same opinion; (ii) polarization, when opinions are clustered around two distinct poles; (iii) fragmentation, when opinions are clustered around more than two poles [28,38].

Moreover, OD models are often categorized based on how opinion variables are defined. In *discrete* opinion models, opinions can take on a limited set of distinct values, and rules dictate changes between them. The Voter Model [9], for instance, focuses on scenarios where individuals can adopt one of two opposing

stances (e.g., agree/disagree). In contrast, *continuous* opinion models represent opinions as points within a continuous spectrum. Classical continuous models, such as the DeGroot Model [14], assume that any opinion within the range [-1, 1] is possible, and interactions between individuals can shift opinions gradually.

LLM-based Social Simulations. Recent research has focused on analyzing the dynamics of interacting LLMs. Generally, LLM agents show believable behaviors both at the individual and collective levels [30], and human-like behaviors naturally emerge from their interactions, e.g., scale-free networks [12], information spreading [18], and trust [42]. When prompted with demographic information and personality traits, they more closely reproduce human behavior [30,31], including human biases and political leaning [6]. The combination of LLMs and traditional Agent-Based modeling [24] as in [40] demonstrated the impact of recommender systems over the quality of the discussion, showing that the exposure of agents to popular contents foster toxicity and interpartisan interaction, reaching levels of toxicity similar to the ones found on tweets by people living in the U.S. in 2019. Additionally, simulations have shown how confirmation bias among agents can lead to fragmentation [6], in line with research on opinion dynamics. LLMs can replicate humans' persuasion dynamics [2,26], producing well-reasoned arguments incorporating psycho-linguistic opinion change theories and can be leveraged for completed social media simulations [34]. However, LLMs have an inherent bias towards accurate information and, unless prompted accurately, avoid producing utterances that conflict with scientific fact [6]. This can cause agents to ignore their assigned personality or role [39], leading to less believable behaviors. LLM agents also tend to avoid harsh conflicts [8,40] and are sensible to several social/political topics [16,35].

3 LLM Opinion Dynamics: A General Framework

In the following, we describe the methodological framework used to simulate LLM discussions and track their opinions. We define our framework for LLM Opinion Dynamics as the following tuple:

$$\mathfrak{M} = \langle \mathcal{G}, \mathcal{S}, \mathcal{O}, f, T, \mathcal{D} \rangle, \tag{1}$$

where:

- $\mathcal{G} = \langle V, E \rangle$ is a graph structure, such that V denotes the set of LLM agents and $E \subseteq V \times V$ denotes the set of unordered agent pairs (i, j) that can engage in a discussion. $\mathcal{G}$ may be a complete graph (i.e., all possible edges exist, a scenario referred to as *mean field*), as well as any real or simulated topology ;
- $\mathcal{S}$ is a statement, namely a claim that is used to initiate the discussion. The statement sets the topic of discussion for the simulation and will serve as a reference for interpreting opinion values.

- $\mathcal{O}$ represents the opinion space, namely the set of possible opinions. Depending on the context, this can be either discrete (e.g., $\mathcal{O} = \{0, 1\}$ for binary opinions) or continuous (e.g., $\mathcal{O} = [0, 1]$ for opinions ranging between 0 and 1). Opinions depend on the topic set by the statement $\mathcal{S}$, and range from perfect disagreement ($min(\mathcal{O})$) to perfect agreement ($max(\mathcal{O})$);
- $f : \mathcal{O}^n \times \mathbb{R}^m \rightarrow \mathcal{O}$ is a function that describes how an agent updates its opinion based on its current opinion, the opinions of its neighbors, and possibly other parameters or external influences. Here, n is the number of neighbors and m represents other influencing factors. Note that in an LLM-powered model, the full form of f is unknown, as it also depends on the LLM's (unknown) parameters.
- T defines the temporal evolution of the system, namely an ordered set of adjacent positive integers $t \in \mathbb{Z}^+$.
- $\mathcal{D}(t) = \{o_i(t) \mid i \in V\}$ is the distribution of opinions over all agents at time t, where $o_i(t) \in \mathcal{O}$ represents the opinion of agent i at time t.

This framework generalizes to all social topologies, interaction scenarios, and discussion topics. In the following, we introduce a specialization of the framework used in the experimental campaign. A general overview of its rationale is provided in Algorithm 1.

Algorithm 1. LLM Opinion Dynamics

Require: Graph: $\mathcal{G}$, Statement: $\mathcal{S}$, Initial opinions: $\mathcal{D}(0) = \{o_i(0) \mid i \in V\}$, Time steps: $T_{\max}$,
Ensure: Opinion Distribution Evolution $\mathcal{D}_{all}$

```
Init D_all as an empty list
D_all.append(D(0))
for t = 0 to T_max − 1 do
    for each agent i ∈ V do
        Select random neighbor v_j ∈ Neighbors(i)
        Δo_i(t) = debate(o_i(t), o_j(t), S)
        Update opinion: o_i(t + 1) = o_i(t) + Δo_i(t)
    end for
    Update opinion distribution: D(t + 1) = {o_i(t + 1) | i ∈ V}
    D_all.append(D(t + 1))
end for
return D_all
```

In detail, we start by defining the graph topology, the initial statement, the opinion distribution, and the number of iterations. Each agent is initialized with a discrete opinion on a 7-point Likert scale, such that 0 corresponds with strong disagreement with S, 3 corresponds with neutrality, and 6 denotes strong agreement. At each time step, for each agent $i \in V$ a neighbor j is randomly selected to engage in a debate on $\mathcal{S}$. As a result of the debate, i's opinion is either (i) brought closer to j's if they agree, (ii) brought farther from j's if they disagree, or (iii) not modified if j's arguments are ignored. Opinion updates are stored at each iteration, and returned at the end.

The debate unfolds according to Algorithm 2. The selected agents are first assigned one of two mutually exclusive roles, namely discussant (agent i) or opponent (agent j). The discussant is the agent who starts the conversation. It

does so by expressing and justifying its opinion on S. The discussant is the only agent allowed to change its opinion (after engaging with the opponent). The opponent is tasked to produce a persuasive decision with the goal of convincing the discussant. It is a stubborn agent, i.e., it is not allowed to change opinion.

The debate proceeds in rounds, where each round consists of one interaction

Algorithm 2. Debate

Require: Initial opinions: o_i, o_j, Statement: $\mathcal{S}$, Maximum number of interaction rounds: `max_rounds`
Ensure: Final opinion delta of the Discussant: $o_i^{\Delta+}$

```
Init Discussant's Opinion Delta o_i^{Δ+} ← 0
Discussant states its opinion o_i on S
for round = 1 to max_rounds = 3 do
    Opponent's Turn:
    Opponent evaluates Discussant's argument
    Opponent decides and justifies: ACCEPT, REJECT, IGNORE
    if decision is IGNORE or ACCEPT then
        break
    end if
    Discussant's Turn:
    Discussant evaluates Opponent's argument
    Discussant decides and justifies: ACCEPT, REJECT, IGNORE
    if decision is ACCEPT then
        o_i^{Δ+} ← 1 if o_j > o_i else -1
        break
    else if decision is REJECT then
        o_i^{Δ+} ← -1 if o_j > o_i else 1
        break
    end if
end for
return o_i^{Δ+}
```

from each agent. The total number of rounds is capped by a maximum limit, `max_rounds`, to ensure the debate does not continue indefinitely. Initially, the Discussant is prompted to express its opinion on $\mathcal{S}$. During each round, the Opponent first evaluates the Discussant's message and then decides whether to `ACCEPT`, `REJECT`, or `IGNORE` the Discussant's viewpoint. If the Opponent chooses to `IGNORE` or `ACCEPT`, the debate concludes. Otherwise, the Discussant evaluates the Opponent's argument, and the debate continues. The final opinion shift of the Discussant, $o_i^{\Delta+}$, is determined based on its final choice. If the Discussant `ACCEPT`s, its opinion is brought closer to the Opponent's, and if it `REJECT`s, it is brought further (e.g., backfire effect [29]). the Opponent's argument. If the Discussant does neither, or if the debate terminates earlier, its opinion remains unchanged.

4 Opinion Dynamics Analysis

Analytical Setting. To study the effects of different node mixing patterns on LLM opinion evolution, we perform simulations on Peel's Quintet [32]. The Quintet is a collection of five graphs characterized by the same number of nodes

(40) and edges (160), the same Nominal Assortativity Coefficient [27], but different local mixing patterns (i.e., assortative/disassortative areas in each graph). Each node is assigned one of two labels (classes), and edges are placed so that the number of connections between and within classes is the same across the quintet. We provide a visual representation in Fig. 1.

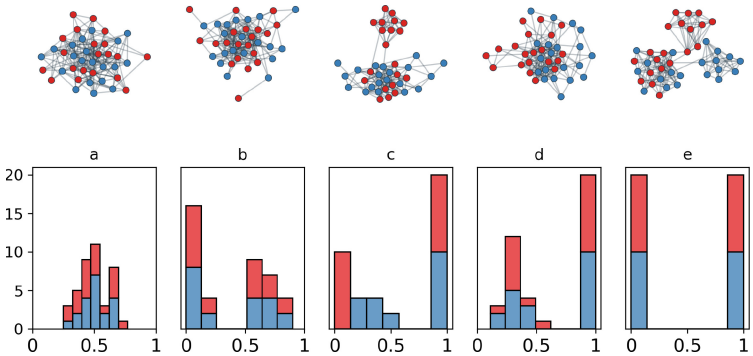


Fig. 1. Peel's Quintet. The top row displays the quintet using the Fruchterman-Reingold layout algorithm [17]. Colors identify node classes. The bottom row displays the distribution of the ratios of same-class neighbors for each node in the corresponding graph (later called *Homogeneity* (Eq. 2.))

The chosen discussion statement refers to the paradox of Theseus' Ship, first recorded in Plutarch's Parallel Lives [15]. In essence, the paradox debates whether the ship did or did not keep its identity after all of its parts were replaced with new ones. The rationale behind this choice is to mitigate LLM bias toward accurate scientific truth, as this causes a faster convergence and reduces the variety of persuasive arguments to support their claims [6]. The input statement in our experiments declares that the ship remains the same. We set red nodes in Fig. 1 to *strongly disagree* with the statement (opinion 0), and blue nodes to *strongly agree* (opinion 6). Since we are also interested in understanding potential behavioral differences among different language models, we employ two LLMs: `Mistral-7B Instruct` (Mistral, henceforth) and `Llama-3-8B` (Llama3) and perform two sets of simulations. In the first set, we set Mistral agents for all red nodes in Fig. 1, and Llama agents for blue ones. In the second set, we set Llama agents for red nodes, and Mistral agents for blue ones. Each simulation is observed for 100 iterations.

In the following, we first analyze opinion trends for both sets of simulations. Then, we study the extent to which a node is surrounded by peers with the same opinion over time. To measure this, we define ego network *homogeneity* as follows:

$$Homogeneity(u) = \frac{|\{v : v \in neighbors(u) \wedge o_u = o_v\}|}{|neighbors(u)|}, \tag{2}$$

where u is the target node with opinion o_u, and v is any of u's neighbors such that they have the same opinion value. All code and data produced in this study are available in a dedicated GitHub repository.[1]

Opinion Trends. Fig. 2 displays opinion trends for both sets of experiments. Panels in the top row refer to the first set, and panels in the bottom one refer to the second. In all configurations, agents tend to accept opponents' opinions and ultimately stabilize on *agree* (5). This effect could be explained by the excessive politeness of LLMs, which are often assertive and tend to avoid conflicts [33]. The most noticeable difference relates to the rate at which negative opinions disappear. In the absence of biases (Figs. 2a and 2f), for instance, opinions take 70–80 iterations to converge. For mostly disassortative networks (Figs. 2b and 2g), convergence is quicker (20–60 iterations). In configurations where nodes show high assortativity, behaviors vary widely. Indeed, the systems in Figs. 2c and 2h, where nodes show high assortativity coupled with some disassortativity, do not converge (in the first simulation) or converge at the very end (in the second experiment). As mixing biases (i.e., both dis/assortative behaviors) become more prominent, systems take 75 iterations or more to converge in the first set of simulations (Figs. 2d and 2e). In the second set, similar behaviors can be noted when assortative and disassortative mixing are equally present (Fig. 2j). However, faster convergence can be observed when assortativity is high, and disassortativity is moderate. In general, in the second set of simulations (where we assign *strongly disagree* opinions to Llama agents and *strongly agree* opinions to Llama ones), we observe relatively faster convergence compared to the first set. Most notably, in Fig. 2g, the bias toward positive opinions leads to the unique emergence (and fast convergence) of *strongly agree*ing agents. We attribute this difference to the inherent biases of LLM models.

Neighborhood Homogeneity. Fig. 3 displays Homogeneity distributions over time for both sets of experiments. Figures 3a-3e show results for the first set of simulations, and Figs. 3f-3j show results for the second. Note that the upper curve in each panel (labeled as `0`) displays the initial homogeneity values, that is the same observed in Fig. 1. Moving downwards, the curves outline the homogeneity distributions every 25 iterations. We observe the tendency of nodes toward local homogeneity, where most nodes show full homogeneity within 100 iterations in all scenarios. What is more, agents show mostly-homogeneous neighborhoods early on. While results in the previous section highlight the tendency of accepting the initial statement, here we also observe the tendency to gradually align with local opinions. When no mixing biases are present (Figs. 3a and 3f) LLM neighborhoods show both strong heterogeneity and homogeneity before converging. Similarly, when both assortative and disassortative behaviors are present (Figs. 3e and 3j), neighborhoods maintain this duality, albeit with less prevalent heterogeneous patterns. In high-assortativity networks (Figs. 3c, 3d, 3h, 3i),

[1] https://github.com/GiulioRossetti/LLM_Opinion_multi_model.

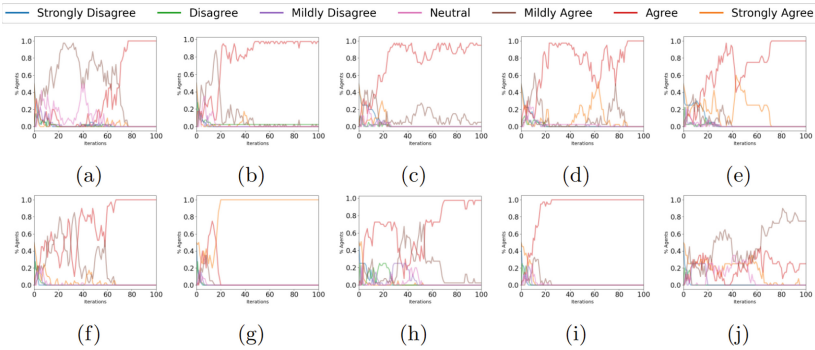


Fig. 2. Percentage of agents holding opinion o as a function of time. Opinions are mapped to the following labels: strongly disagree, disagree, mildly disagree, neutral, mildly agree, agree, and strongly agree. The top panels refer to simulations where Mistral agents are assigned the red class, and Llama agents are assigned the blue class. The bottom panels refer to the opposite scenario.

instead, homogeneous neighborhoods dominate the social landscape, and only a negligible fraction show heterogeneous patterns before disappearing.

5 Discussion and Conclusions

In this paper, we formalized a framework for studying Opinion Dynamics of Large Language Models and evaluated how different local node mixing patterns impact the evolution of opinions. We observe that, in most cases, most agents tend to converge towards agreeing with the initial statement within the first 100 iterations, regardless of the extreme views in the initial state. Furthermore, upon reaching convergence, the agents do not adopt the most extreme opinion (6) but maintain a moderate one (5). We also find that mostly-disassortative LLM societies show faster convergence with respect to cases where no mixing bias is present. However, the joint effect of assortative and disassortative patterns seems to lead to even slower/no convergence. We also highlight that LLM opinion dynamics are not oblivious to initial LLM configurations: in our simulations, assigning *strongly agree* opinions to Llama and *strongly disagree* opinions to Mistral agents leads to comparatively faster convergence than the opposite scenario. Finally, the homogeneity analysis unveiled LLMs' tendency to gradually align with local opinions, especially when the initial conditions are already assortative. Future work could replicate our study on larger populations or in more realistic simulation environments [34], investigating the role of other network properties in shaping LLMs' opinions, such as network density or different degree distributions. Moreover, by leveraging LLMs' textual productions, future works could also study language dynamics and whether/how they are affected by different network configurations.

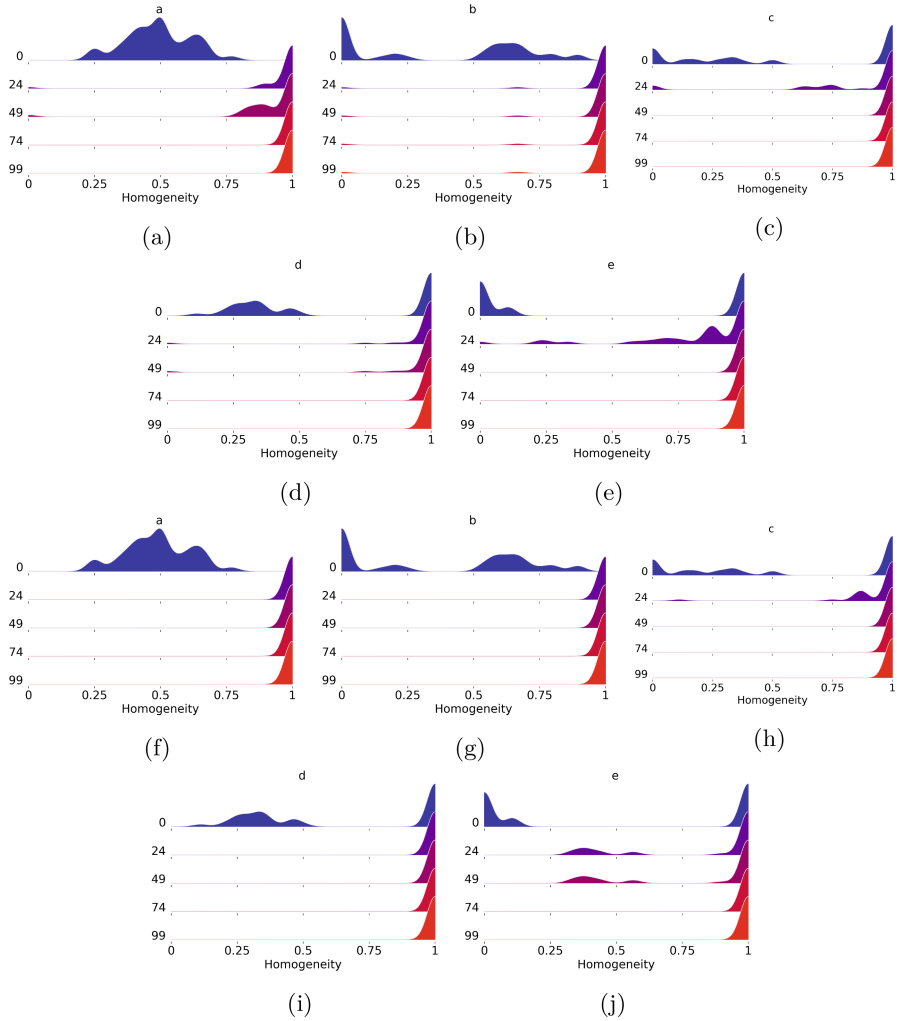


Fig. 3. Kernel Density Estimation of Homogeneity values over time for the first (a-e) and second (f-j) sets of simulations. In each panel, values are shown every 25 iterations, with the top curve being the starting condition (i.e., the same setting as Fig. 1, bottom row).

Acknowledgements. This work is supported by (i) the European Union – Horizon 2020 Program under the scheme "INFRAIA-01-2018-2019 – Integrating Activities for Advanced Communities", Grant Agreement n.871042,"SoBigData++: European Integrated Infrastructure for Social Mining and Big Data Analytics" (http://www.sobigdata.eu); (ii) SoBigData.it which receives funding from the European Union – NextGenerationEU – National Recovery and Resilience Plan (Piano Nazionale di Ripresa e Resilienza, PNRR) – Project: "SoBigData.it – Strengthening the Italian RI for Social Mining and Big Data Analytics" – Prot. IR0000013 – Avviso n. 3264

del 28/12/2021; (iii) EU NextGenerationEU programme under the funding schemes PNRR-PE-AI FAIR (Future Artificial Intelligence Research).

References

1. Aroyehun, S.T., Malik, L., Metzler, H., Haimerl, N., Di Natale, A., Garcia, D.: LEIA: linguistic embeddings for the identification of affect. EPJ Data Sci. **12**(1), 52 (2023)
2. Breum, S.M., Egdal, D.V., Mortensen, V.G., Møller, A.G., Aiello, L.M.: The persuasive power of large language models. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 18, pp. 152–163 (2024)
3. Brown, T., et al.: Language models are few-shot learners. Adv. Neural. Inf. Process. Syst. **33**, 1877–1901 (2020)
4. Brunet-Gouet, E., Vidal, N., Roux, P.: Do conversational agents have a theory of mind? A single case study of ChatGPT with the hinting, false beliefs and false photographs, and strange stories paradigms (2023)
5. Castellano, C., Fortunato, S., Loreto, V.: Statistical physics of social dynamics. Rev. Mod. Phys. **81**(2), 591–646 (2009)
6. Chuang, Y.S., et al.: Simulating opinion dynamics with networks of LLM-based agents. arXiv preprint arXiv:2311.09618 (2023)
7. Chuang, Y.S., et al.: The wisdom of partisan crowds: comparing collective intelligence in humans and LLM-based agents. In: Proceedings of the Annual Meeting of the Cognitive Science Society, vol. 46 (2024)
8. Cisneros-Velarde, P.: On the principles behind opinion dynamics in multi-agent systems of large language models. arXiv preprint arXiv:2406.15492 (2024)
9. Clifford, P., Sudbury, A.: A model for spatial conflict. Biometrika **60**(3), 581–588 (1973). https://doi.org/10.1093/biomet/60.3.581
10. Crespi, I.: The Public Opinion Process: How the People Speak. Routledge (2013)
11. Cuzzolin, F., Morelli, A., Cîrstea, B., Sahakian, B.J.: Knowing me, knowing you: theory of mind in AI. Psychol. Med. **50**(7), 1057–1061 (2020). https://doi.org/10.1017/s0033291720000835
12. De Marzo, G., Pietronero, L., Garcia, D.: Emergence of scale-free networks in social interactions among large language models. arXiv preprint arXiv:2312.06619 (2023)
13. Deffuant, G., Neau, D., Amblard, F., Weisbuch, G.: Mixing beliefs among interacting agents. Adv. Complex Syst. **3**(01n04), 87–98 (2000)
14. Degroot, M.H.: Reaching a consensus. J. Am. Stat. Assoc. **69**(345), 118–121 (1974). https://doi.org/10.1080/01621459.1974.10480137
15. Dryden, J., Clough, A.: Parallel Lives. e-artnow (2018). https://books.google.it/books?id=-amSDwAAQBAJ
16. Farina, M., Lavazza, A.: ChatGPT in society: emerging issues. Front. Artif. Intell. **6** (2023). https://api.semanticscholar.org/CorpusID:259168081
17. Fruchterman, T.M., Reingold, E.M.: Graph drawing by force-directed placement. Softw. Pract. Experience **21**(11), 1129–1164 (1991)
18. Gao, C., et al.: S3: social-network simulation system with large language model-empowered agents. arXiv preprint arXiv:2307.14984 (2023)
19. Holley, R.A., Liggett, T.M.: Ergodic theorems for weakly interacting infinite systems and the voter model. Ann. Probab. **3**(4), 643–663 (1975)
20. Hu, T., Collier, N.: Quantifying the persona effect in LLM simulations. arXiv preprint arXiv:2402.10811 (2024)

21. Kosinski, M.: Theory of mind might have spontaneously emerged in large language models. arXiv preprint arXiv:2302.02083 (2023)
22. Krapivsky, P.L., Redner, S.: Dynamics of majority rule in two-state interacting spin systems. Phys. Rev. Lett. **90**(23), 238701 (2003)
23. Lan, X., Gao, C., Jin, D., Li, Y.: Stance detection with collaborative role-infused LLM-based agents. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 18, pp. 891–903 (2024)
24. Lorig, F., Johansson, E., Davidsson, P.: Agent-based social simulation of the COVID-19 pandemic: a systematic review. J. Artif. Soc. Soc. Simul. **24**(3) (2021)
25. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: homophily in social networks. Ann. Rev. Sociol. **27**(1), 415–444 (2001)
26. Monti, C., Aiello, L.M., De Francisci Morales, G., Bonchi, F.: The language of opinion change on social media under the lens of communicative action. Sci. Rep. **12**(1), 17920 (2022)
27. Newman, M.E.: Mixing patterns in networks. Phys. Rev. **67**(2), 026126 (2003)
28. Noorazar, H.: Recent advances in opinion propagation dynamics: a 2020 survey. Eur. Phy. J .Plus **135**, 1–20 (2020)
29. Nyhan, B., Reifler, J.: When corrections fail: the persistence of political misperceptions. Polit. Behav. **32**(2), 303–330 (2010)
30. Park, J.S., O'Brien, J., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative agents: interactive simulacra of human behavior. In: Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, pp. 1–22 (2023)
31. Park, J.S., Popowski, L., Cai, C., Morris, M.R., Liang, P., Bernstein, M.S.: Social simulacra: creating populated prototypes for social computing systems. In: Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology, UIST '22. Association for Computing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3526113.3545616
32. Peel, L., Delvenne, J.C., Lambiotte, R.: Multiscale mixing patterns in networks. Proc. Natl. Acad. Sci. **115**(16), 4057–4062 (2018)
33. Priya, P., Firdaus, M., Ekbal, A.: Computational politeness in natural language processing: a survey. ACM Comput. Surv. **56**(9), 1–42 (2024)
34. Rossetti, G., et al.: Y social: an LLM-powered social media digital twin. arXiv preprint arXiv:2408.00818 (2024)
35. Rozado, D.: The political preferences of LLMs. PLOS ONE **19** (2024). https://api.semanticscholar.org/CorpusID:267412830
36. Schweitzer, F.: Sociophysics. Phys. Today **71**(2), 40–46 (2018)
37. Si, X.M., Li, C.: Bounded confidence opinion dynamics in virtual networks and real networks. J. Comput. **29**(3), 220–228 (2018)
38. Sîrbu, A., Loreto, V., Servedio, V.D., Tria, F.: Opinion dynamics: models, extensions and external effects. In: Participatory Sensing, Opinions and Collective Awareness, pp. 363–401 (2017)
39. Taubenfeld, A., Dover, Y., Reichart, R., Goldstein, A.: Systematic biases in LLM simulations of debates. arXiv preprint arXiv:2402.04049 (2024)
40. T"ornberg, P., Valeeva, D., Uitermark, J., Bail, C.: Simulating social media using large language models to evaluate alternative news feed algorithms. arXiv preprint arXiv:2310.05984 (2023)
41. Wei, J., et al.: Emergent abilities of large language models. arXiv preprint arXiv:2206.07682 (2022)
42. Xie, C., et al.: Can large language model agents simulate human trust behaviors? arXiv preprint arXiv:2402.04559 (2024)

Risk-Taking Influences the Network Growth and Population Dynamics in a Spatial Prisoner's Dilemma

Tomoko Sakiyama(✉) and Akihiro Takahara

Soka University, 1-236, Tangi-Cho, Tokyo, Japan
sakiyama@soka.ac.jp

Abstract. In this paper, we develop a spatial prisoner's dilemma model by considering the memory use of players and apply it to a network system to evolve network dynamics using a SPD model. In our model, players sometimes ignore the classical strategy update rule and adopt a strategy that is unlikely to occur. Furthermore, players sometimes connect to or disconnect from another player if their situation is not favorable. Thanks to these effects, our model presents dynamic network evolutions with a mixture of cooperative and defective populations.

Keywords: SPD · Network growth · Cooperative evolution

1 Introduction

Cooperative behaviors are a defining feature of mammals and other living systems [1]. At the same time, selfish behaviors are also common in these systems. The study of spatial game theories, including the spatial prisoner's dilemma (SPD), has been widely investigated to reveal the evolution of cooperative behaviors in selfish populations [1–3]. In SPD, players are assigned one of two strategies: defector or cooperator. If a defector interacts with a cooperator, only the defector wins. On the other hand, if a cooperator interacts with another cooperator, they establish a win-win relationship. As defectors earn higher payoffs from cooperators, cooperative populations tend to vanish from the system [4, 5]. To overcome this problem, numerous SPD models have been developed [6–8]. Several studies have focused on the memory use of players [8–10], as biological systems often rely on some form of memory.

It seems that the long-term memory capacity of players enables the system to evolve cooperative behaviors. However, it is not always reasonable for players to assume that they can access a long history, as some types of living systems cannot retain their memory for extended periods. Therefore, the aim of this study is to develop an SPD model where players can only access recent memory.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 177–184, 2025.
https://doi.org/10.1007/978-3-031-82439-5_15

In addition, we also focus on network growth using this SPD model. Recently, spatial game theories have been applied to network systems to investigate their growth [11, 12]. In fact, real-world networks are heterogeneous, like scale-free networks. Therefore, we introduce link connection and disconnection in our SPD model and evaluate network growth and its characteristic properties.

In our model, players with lower scores are likely to connect to or disconnect from another player in order to improve their situation and earn higher scores.

2 Model Description

2.1 Space and Agent

Here, we set up a simulation with 100 players. At the beginning of each trial, players were randomly connected to each other. The average degree of nodes was set to 5. Each player was randomly assigned one of two strategies: defector or cooperator. The simulation time for each trial lasted for 1000 time steps.

2.2 Perverse Prisoner's Dilemma (PPD) with Network Growth Model

In this paper, we apply PPD model to the network growth [11 and 12]. First, we explain the PPD sub-model and then we explain the sub models of link connection/disconnection in our model.

Table 1. A payoff matrix

Strategies	D	C
D	b	0
C	0	1

In the PPD on a network, players earn scores by competing with all their adjacent players using a payoff matrix (See Table 1). Here, the parameter b satisfies $1 < b < 2$. Then, individuals compare their calculated scores with those of their adjacent players. Players update their strategies to match those of the adjacent player who earns the highest score among their neighbors. Thus, the basic process of the PPD model is similar to that of classical SPD models [4]. However, players sometimes ignore the classical strategy update rule and adopt another method as follows: If two or more adjacent players earn the highest score among their neighbors but their strategies are not all the same, player i uses its memory to update its strategy.

The player tallies the number of cooperators among themselves and their neighboring players at the current time (t) and at one previous time: $Count^t(i)$ and $Count^{t-1}(i)$. Then, the following equation is applied to the player.

With a probability $\frac{\mathrm{Count^t(i)+Count^{t-1}(i)}}{\mathrm{n^t(i)+n^{t-1}(i)}}$,

the strategy for player i is replaced with a defector.

With a probability $1 - \frac{\mathrm{Count^t(i)+Count^{t-1}(i)}}{\mathrm{n^t(i)+n^{t-1}(i)}}$,

the strategy for player i is replaced with a cooperator.

Here, $n^t(i)$ presents the total number of itself and adjacent members for player i at time t.

Then, we explain the sub-model of the network growth in our model. In our model, player i connects/disconnects to/from one player whenever it earns the lowest score among its neighbors as follows:

With a probability p,

player i randomly selects one player who is not an adjacent player for player i but who earns the highest score among the player k's neighborhood, which is one of adjacent player for player i. Note that player k is randomly selected.

With a probability $1 - \mathrm{p}$,

player i randomly disconnects its link from one of its adjacent players who earning the highest score among its neighborhoods.

Therefore, at one time, players are likely to connect to an unknown player who is a friend with their friend (1-neighborhood of the adjacent players). At another time, it is likely to stop being friends with one of their friends.

3 Results

Here, we set parameters as shown in Table 2 for simulations.

Table 2. Parameters

Parameter Name	Values
Number of players	100
Initial defector density	0.5
Initial degree of nodes	5 (averaged)
Simulation time	1000

First, we evaluated the parameter p effects. Figure 1 shows the average node degree at the end of trials obtained from 100 trials. As shown in this figure, the degree of nodes can be evolved as the parameter p increases. Figure 2 demonstrates examples of fraction of defectors for several parameter values. These results confirm that high parameter p values achieve the evolutions of cooperators.

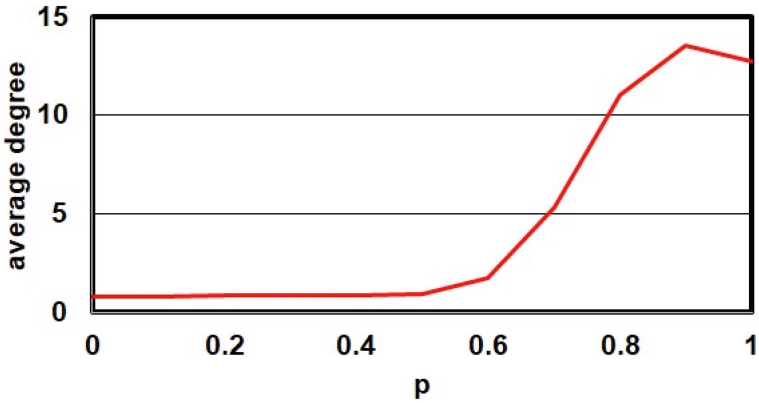


Fig. 1. Averaged degree of nodes at the end of simulations obtained from 100 trials.

This is perhaps because cooperative players can connect with each other as parameter p increases. If this parameter is low, players cannot connect with each other, and almost all players become defectors (Fig. 2A). Therefore, the process of link connections is essential for players to evolve populations of cooperators. However, at the same time, defectors are likely to be connected with opponent players. Consequently, several defectors can earn the highest scores by being connected to more opponent players. These processes are related to the growth of hub nodes. We examined the relationship between the degree of nodes at the end of simulations and players' strategies. Figures 3 and 4 present the results for $p = 0.70$ and 0.90, respectively. As observed in these figures, some players become hub nodes. Interestingly, defectors are more likely to become hub nodes, although cooperators can also become hub nodes to some extent (Fig. 3).

Finally, we examined the growth of node degrees for players. Figure 5 shows examples of the time evolution of node degrees using $p = 0.70$. Nodes sometimes decrease their degree even after being hub nodes for a while (Fig. 5A). However, some of them later regain hub status (Fig. 5B). Therefore, hub players can be replaced by other players. Our model demonstrates dynamic evolutions of node degrees, which are also observed in real network systems [11].

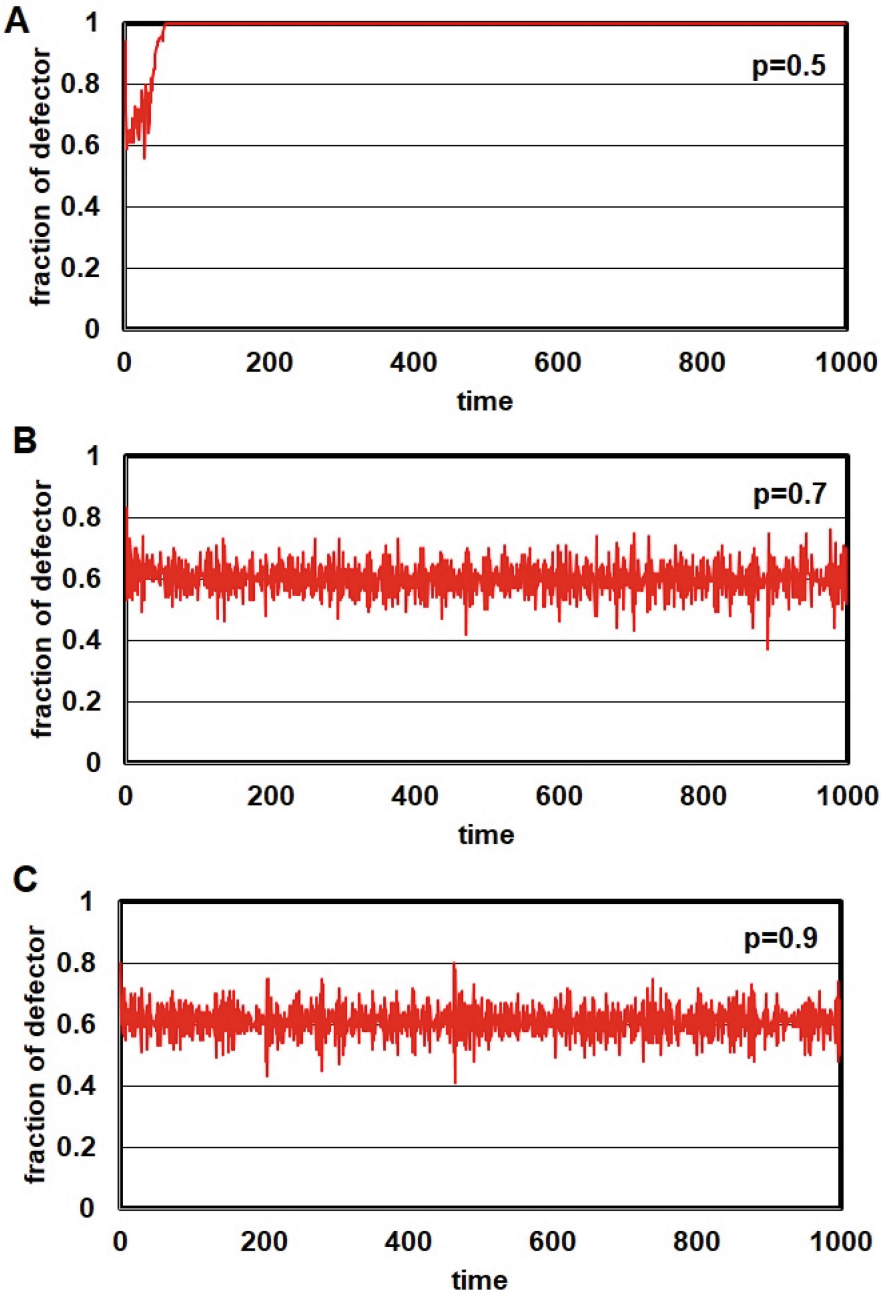


Fig. 2. Fraction of defectors as time goes on. A. $p = 0.70$. B. $p = 0.80$. C. $p = 0.90$.

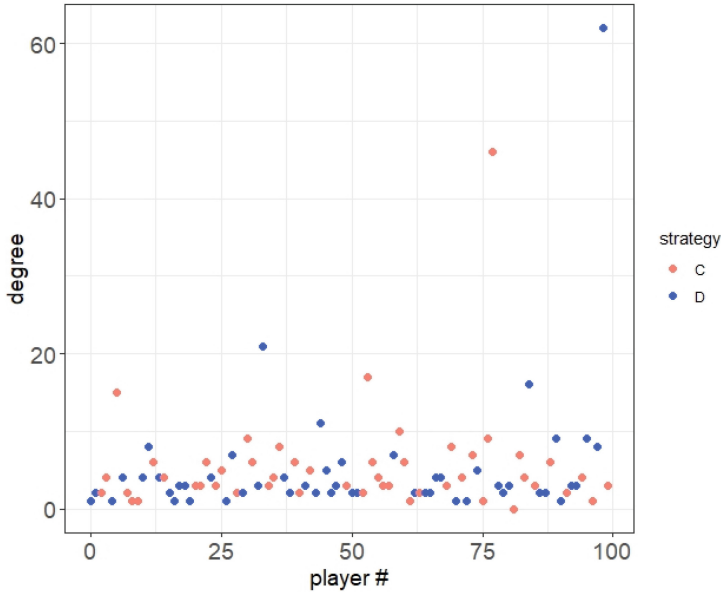


Fig. 3. Node degree of individual players after 1trial with $p = 0.70$.

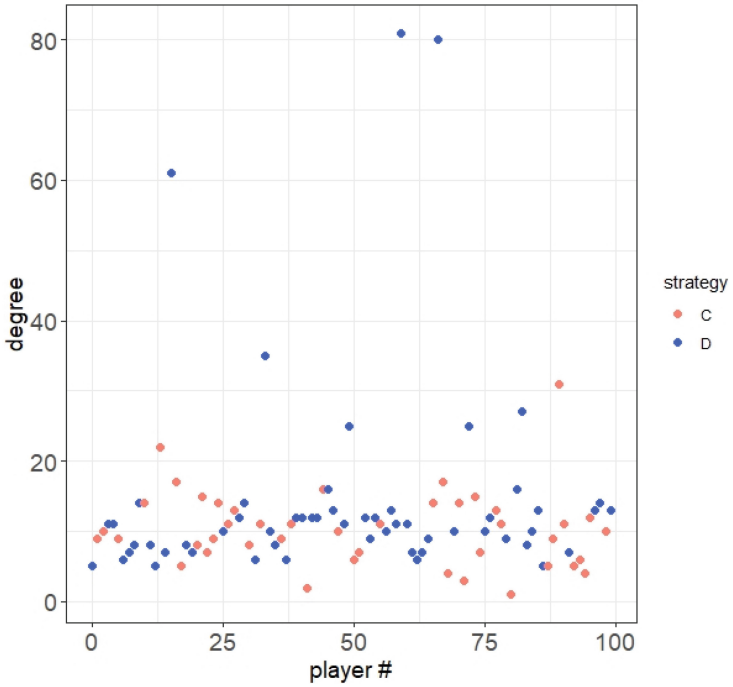


Fig. 4. Node degree of individual players after 1trial with $p = 0.90$.

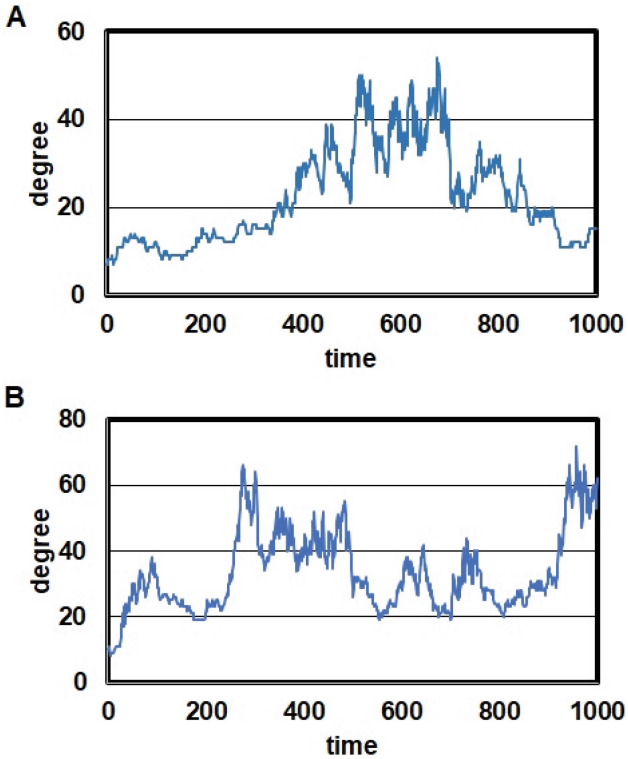


Fig. 5. Node degree and its evolutions for two players with $p = 0.70$. A. The first player. B. The second player. Two players were selected randomly.

4 Conclusion

In this paper, we investigated the influence of risk-taking in SPD and found that it affects the dynamical evolution of populations and the emergence of heterogeneous properties in network growth. In our model, players earning lower scores are likely to ignore the classical strategy update rule and adopt a non-optimal strategy update rule. This is because players in our model sometimes adopt a strategy that is unlikely to occur. In that sense, players take risks and behave differently from before. Furthermore, players earning the worst scores in their neighborhood are likely to change their adjacency by connecting or disconnecting with another player. Changing neighborhood conditions does not necessarily result in a better situation for players. In that sense, players also take risks regarding the evolution of links.

At first glance, risk-taking may seem unreasonable for players. However, our findings suggest that these non-optimal decision-making processes can contribute to the dynamical evolution of SPD network systems.

References

1. Axelrod, A.: The Evolution of Cooperation. Basic Books, New York (1984)
2. Nowak, M., May, R.: Evolutionary games and spatial chaos. Nature **359**, 826–829 (1992). https://doi.org/10.1038/359826a0
3. Hauert, M., Doebeli, M.: Spatial structure often inhibits the evolution of cooperation in the snowdrift game. Nature **428**(6983), 643–646 (2004)
4. Killingback, T., Coebeli, M.: Spatial evolutionary game theory: hawks and doves revisited. Proc. R. Soc. Lond. B **263**(1374), 1135–1144 (1996)
5. Maynard Smith, J., Price, G.: The logic of animal conflict. Nature **246**, 15–18 (1973)
6. Qin, J., Chen, Y., Fu, W., Kang, Y., Perc, M.: Neighborhood diversity promotes cooperation in social dilemmas. IEEE Access **6**, 5003–5009 (2018)
7. Kojo, K.I., Sakiyama, T.: Restructuring of neighborhood definition based on strategies will enhance the cooperation in a spatial prisoner's dilemma. Chaos Solitons Fractals **179**, 114404 (2024)
8. Danku, Z., Perc, M., Szolnoki, A.: Knowing the past improves cooperation in the future. Sci. Rep. **9**, 262 (2019). https://doi.org/10.1038/s41598-018-36486-x
9. Deng, Z., Ma, C., Mao, X., Wang, S., Niu, Z., Gao, L.: Historical payoff promotes cooperation in the prisoner's dilemma game. Chaos Soliton Fract **104**, 1–5 (2017)
10. Takahara, A., Sakiyama, T.: Twisted strategy may enhance the evolution of cooperation in spatial prisoner's dilemma. Physica A **629**, 129212 (2023)
11. Sakiyama, T.: A power law network in an evolutionary hawk–dove game. Chaos Solitons Fractals **146**, 110932 (2021)
12. Li, D., Ma, J., Han, D., Sun, M., Tian, L., Stanley, H.E.: The co-evolution of networks and prisoner's dilemma game by considering sensitivity and visibility. Sci. Rep. **7**(1), 45237 (2017)

Infrastructure Networks

Evaluating Node Contribution in Network Path Transport Using *Physarum* Centrality

Kiri Kawato[1(✉)], Tetsuya Yamamoto[1(✉)], and Masaki Aida[2]

[1] Tokyo Metropolitan College of Industrial Technology, 1–10–40 Higashioi, Shinagawa, Tokyo 140–0011, Japan
s23013@g.metro-cit.ac.jp, tyama@metro-cit.ac.jp

[2] Graduate School of Systems Design, Tokyo Metropolitan University, 6–6 Asahigaoka, Hino, Tokyo 191–0061, Japan
aida@tmu.ac.jp

Abstract. Maintaining stable transportation in infrastructure networks is a critical issue, even with limited resources and costs. This study investigates the effectiveness of *Physarum* centrality (C_{P}) in measuring each node's contribution to transportation between network nodes. We calculated conductance, a measure of transport efficiency, in Unit Disk Graphs as nodes were removed in ascending order of centrality. Results show that C_{P} maintained higher conductance compared to weighted degree and betweenness centrality, regardless of the node removal rate. C_{P} also outperformed other centralities across varying total transportation flows. Thus, by using C_{P} as an indicator, stable transportation can be maintained with fewer intermediate nodes.

Keywords: *Physarum* Centrality · *Physarum* Algorithm · Centrality Measures · Weighted Networks · Transport Efficiency

1 Introduction

In infrastructure networks such as power grids, transportation systems, and communication networks, maintaining stable transportation is a critical issue. On the other hand, constraints such as capital investment and physical resources need to be addressed. In cases where resource constraints reduce redundancy, cascading failures might occur during emergencies, such as disasters, due to overload or the propagation of faults. To avoid overload and construct networks that suppress such issues, it is effective to quantitatively evaluate each node's contribution to network path transport.

One of the evaluation indicators for nodes within a network is the centrality measure. Degree and betweenness centrality are commonly used to evaluate attack resistance against hubs or connections between clusters in a network. In this study, we focused on *Physarum* centrality [1], which utilized the *Physarum*

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 187–195, 2025.
https://doi.org/10.1007/978-3-031-82439-5_16

algorithm [2], as a metric for evaluating nodes' contributions to network path transport.

The *Physarum* algorithm, proposed by Tero et al. [2], is inspired by the transport network formed by *Physarum polycephalum*. In this model, the organism's transport tubes thicken as the protoplasmic flow increases. The algorithm effectively constructs networks that balance cost, efficiency, and fault tolerance. Leveraging such characteristics, the effectiveness of the *Physarum* algorithm in railway network design has been demonstrated by Watanabe et al. [3] and Houbraken et al. [4].

Physarum centrality, proposed by Zhang et al. [1], is a metric calculated based on the flow derived from applying the *Physarum* algorithm. Degree centrality overlooks the structural importance of a node, while betweenness centrality requires global information about the network. Therefore, both measures have limitations in certain applications. To address the characteristic of betweenness centrality that considers only the shortest paths, extensions that account for propagation along non-shortest paths have been proposed [5,6]. *Physarum* centrality has also been reported as a centrality measure that can take into account the overall structure of the network [1].

This characteristic has led to its use in various research fields, such as the priority immunization of highly infectious nodes to suppress the spread of infection [7], and the identification of critical brain network hubs and bridges [8]. We investigated the effectiveness of *Physarum* centrality, which captures the overall network structure, as a metric for evaluating each node's contribution to network path transport.

2 Centrality Measures

2.1 *Physarum* Centrality

The *Physarum* algorithm proposed by Tero et al. [2] is a mathematical model of the transport tube network formed by *Physarum polycephalum*. Q_{ij} represents the flow rate through the link between nodes i and j. Assuming approximate Poiseuille flow, Q_{ij} is given by the following equation:

$$Q_{ij} = \frac{D_{ij}}{L_{ij}}(p_i - p_j), \tag{1}$$

where L_{ij} is the length of the link between nodes i and j, D_{ij} is its conductivity, and p_i is the pressure at node i.

The inflow and outflow Q_{ij} at each node are related by the following equation:

$$\sum_i Q_{ij} = \begin{cases} -I_0, & \text{for } j = \mathrm{s}, \\ +I_0, & \text{for } j = \mathrm{d}, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where I_0 is the total flow from the source node s to the destination node d, which is constant throughout the network.

Next, the link D_{ij} changes over time according to the function $f(|Q_{ij}|)$ as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t} D_{ij} = f(|Q_{ij}|) - \alpha D_{ij}, \tag{3}$$

where α is the damping term. In this study, we employed a sigmoid function for $f(|Q_{ij}|)$ to constrain the value of D_{ij} between 0 and 1, with $\alpha = 1$. It has been reported [2] that the formed path varies depending on the setting of total flow I_0. When I_0 is low, only the shortest path remains. In contrast, a higher I_0 results in the formation of detour paths as well.

Physarum centrality proposed by Zhang et al. [1] is a centrality measure based on the flow of links obtained from the *Physarum* algorithm. *Physarum* centrality $C_{\mathrm{P}}(i)$ of a node i is calculated using the flow Q_{ij} of the links associated with the node, according to the following formula:

$$C_{\mathrm{P}}(i) = \sum_{j \in \partial i} |Q_{ij}|. \tag{4}$$

In Zhang et al.'s study, $C_{\mathrm{P}}(i)$ was calculated for all nodes in a social network service, assuming interactions between every pair of nodes. In contrast, this study calculated under the assumption that s and d were fixed, similar to a power grids scenario.

2.2 Other Centrality Measures

Weighted degree centrality (C_{D}) is defined as the sum of the weights of the links connected to a node i. For a node i, $C_{\mathrm{D}}(i)$ is defined using the weight w_{ij} of the link between nodes i and j as follows:

$$C_{\mathrm{D}}(i) = \sum_{j \in \partial i} w_{ij}. \tag{5}$$

Weighted betweenness centrality (C_{B}) measures how often a node appears on the shortest paths between other pairs of nodes in the network. This measure considers link weights and uses the weighted shortest paths, defined as the paths with the highest weights. $C_{\mathrm{B}}(i)$ of node i is defined as follows:

$$C_{\mathrm{B}}(i) = \sum_{\substack{j,k \in V \setminus \{i\} \\ j \neq k}} \frac{\sigma_{jk}(i)}{\sigma_{jk}}. \tag{6}$$

Here, V represents the set of nodes, σ_{jk} is the number of shortest paths from node j to node k, and $\sigma_{jk}(i)$ is the number of those shortest paths that pass through node i. In this study, Dijkstra's algorithm was employed to calculate the weighted shortest paths.

3 Simulation on Unit Disk Graphs

3.1 Application of the *Physarum* Algorithm

We modeled the network as a Unit Disk Graph, similar to how power grids might be arranged with power towers or utility poles placed at equal distances. In this model, 40 nodes were considered as power towers and were randomly placed at integer coordinates. Nodes within a radius of 3 were connected by links. The resulting Unit Disk Graph is shown in Fig. 1(a).

Next, a numerical simulation applying the *Physarum* algorithm was conducted with a total flow of $I_0 = 2$ for the network shown in Fig. 1(a). The source node s was set to node 23, and the destination node d was set to node 28, which are the nodes with the maximum hop distance. The geometric distance between nodes was used as the link length L_{ij}, with an initial conductivity of $D_{ij} = 0.5$. Figure 1(b) shows the visualization of the link conductivity D_{ij} at equilibrium. In Fig. 1(b), the links forming the shortest path between s and d are thicker, while the terminal links that are not part of this path are thinner.

Figure 1(c) shows the results obtained by applying the same procedure to simulations with $I_0 = 10$. Comparing Figs. 1(b) and 1(c), it is evident that there are differences in link conductivity between the case with a low total flow, $I_0 = 2$, and the case with a high total flow, $I_0 = 10$. This result indicates that as the flow within the network increased, the transport, which was limited to only the shortest path for $I_0 = 2$, began to distribute across more paths, resulting in an increase in the number of thicker links.

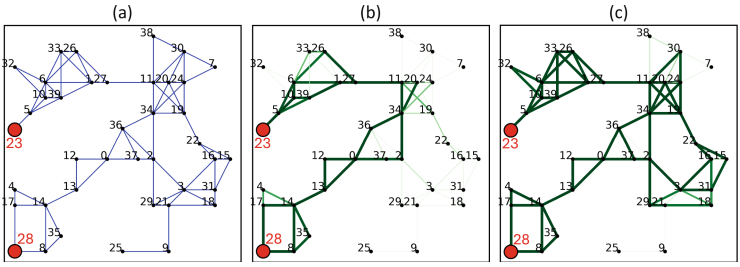


Fig. 1. Results of applying the *Physarum* algorithm to the Unit Disk Graph. The points highlighted in red in the figure are the source node and the destination node. (a) Generated Unit Disk Graph. (b) Visualization of link conductivity D_{ij} after applying the *Physarum* algorithm with $I_0 = 2$. The links with larger D_{ij} values were represented by thicker and darker lines. (c) Results for $I_0 = 10$.

3.2 Comparison of Node Centrality

Next, we calculated the centrality of each node for the network shown in Fig. 1(b) with $I_0 = 2$. First, *Physarum* centrality C_{P} was calculated based on the link

flow Q_{ij} obtained using the *Physarum* algorithm. Subsequently, using $w_{ij} = D_{ij}/L_{ij}$ as the weight, the weighted degree centrality $C_{\mathrm{D}}, C_{\mathrm{DP}}$, and the weighted betweenness centrality $C_{\mathrm{B}}, C_{\mathrm{BP}}$ were calculated. For C_{D} and C_{B}, we used a constant link conductivity of $D_{ij} = 1$. On the other hand, C_{DP} and C_{BP} were calculated using D_{ij} obtained from the *Physarum* algorithm. Table 1 shows the top 5 nodes for each centrality measure.

Table 1. Top 5 nodes with highest centrality measures and their values ($I_0 = 2$)

	C_{D}		C_{DP}		C_{B}		C_{BP}		C_{P}	
Rank	Node	Value	Node	Value	Node	Value	Node	Value	Node	Value
1	6	3.74	10	2.78	11	0.43	34	0.49	14	2.00
2	10	3.41	6	2.43	34	0.40	11	0.45	13	2.00
3	11	3.37	11	2.33	27	0.36	27	0.37	0	2.00
4	20	3.34	0	2.11	0	0.33	0	0.34	27	2.00
5	24	3.30	34	2.07	1	0.27	2	0.30	11	2.00

In Table 1, nodes with high values of C_{D} and C_{DP} include nodes 6 and 10, which are nodes with a higher number of links in the network shown in Fig. 1(b). On the other hand, nodes with high values of C_{B} and C_{BP} include nodes 34 and 27, which serve as bridging nodes between clusters. Nodes such as 11 and 0, which appear high on both weighted degree and betweenness centrality lists, have many links and connect different clusters.

Nodes 0, 27, and 11 also show high values (2.00) for C_{P}. Additionally, nodes 14 and 13 show high C_{P} values as well. These nodes act as bridges to clusters that include the destination node 28 shown in Fig. 1(b), serving as points where transport is concentrated. If nodes with high C_{P} values, such as nodes 14, 13, 0, 27, and 11, were removed, the flow within the network would be disrupted in all cases. However, if only nodes with lower C_{P} values, such as nodes 6, 10, and 34, were removed, the influence would be less severe because detour paths would still be available. This was suggested that C_{P} is an indicator for evaluating the contribution of nodes to maintaining transportation along the s–d paths.

3.3 Transport Efficiency with Node Removal Based on Centrality

Next, we investigated the relationship between each centrality measure and the contribution of nodes to transportation.

The robustness index R is widely used as a metric for evaluating network connectivity against attacks [9]. R evaluates how well network connectivity is maintained as nodes are sequentially removed based on centrality. Inspired by this concept, we assess how removing nodes based on their centrality affects the ease of flow G_{sd} along the s–d path. G_{sd} represents the conductance of the path connecting nodes s and d. It is calculated using the total flow I_0 and the

pressure difference between s and d, denoted as $p_s - p_d$, according to the formula $G_{sd} = I_0/(p_s - p_d)$. This was used as the measure of transport efficiency to evaluate stable transportation along network paths.

Using the network from Fig. 1(b) with $I_0 = 2$, node s was set as node 23, and node d was set as node 28. Figure 2 shows the results of G_{sd} based on the removal of nodes in ascending order of the centrality calculated in Sect. 3.2. The horizontal axis shows the node removal rate, while the vertical axis shows G_{sd} normalized by the value at a removal rate of 0. Note that nodes s and d were not removed during the node removal process.

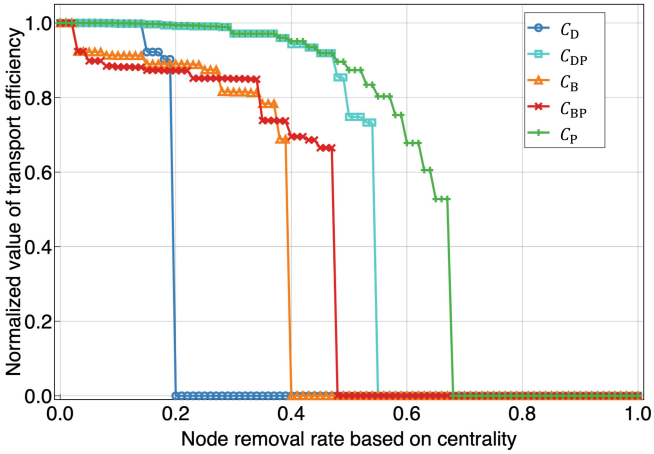


Fig. 2. Transport efficiency G_{sd} when nodes are removed in ascending order of centrality (Number of nodes: 40, single network pattern, $I_0 = 2$). Green lines represent node removal based on C_P, blue lines for C_D, light blue lines for C_{DP}, orange lines for C_B, and red lines for C_{BP}.

The results shown in Fig. 2 indicate that for all centralities, G_{sd} decreases as the number of removed nodes increases. For C_P, G_{sd} remains close to 1 up to a removal rate of 0.3, but drops to $G_{sd} = 0$ once the removal rate exceeds 0.7. The order in which G_{sd} reaches 0 as the node removal rate increases is $C_D < C_B < C_{BP} < C_{DP} < C_P$. Among the calculated centralities, C_P is the least affected in terms of G_{sd} reduction. Thus, it was shown that C_P evaluates the contribution of nodes to maintaining transportation paths.

Figure 2 shows results based on a single network pattern. To reduce bias, the number of nodes was increased to 100, and similar simulations were conducted on 20 different network patterns. The average results are shown in Fig. 3.

Comparing the results in Fig. 3 with those in Fig. 2, increasing the number of network patterns led to a smoother decrease in G_{sd} rather than an abrupt drop. The centrality with the smallest reduction in G_{sd} remains C_P, confirming its superior performance, as was the case in Fig. 2.

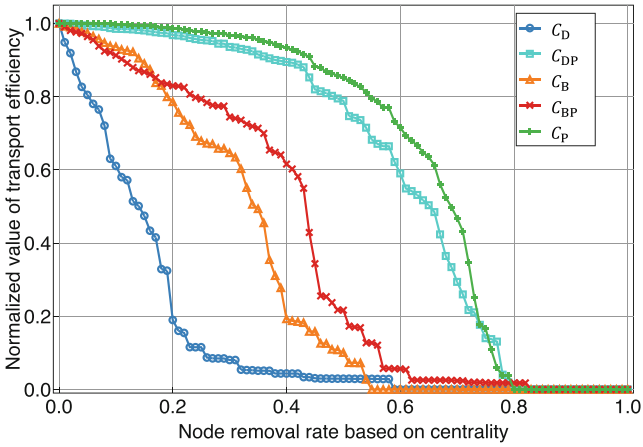


Fig. 3. Transport efficiency G_{sd} when nodes are removed in ascending order of centrality (Number of nodes: 100, average of 20 network patterns, $I_0 = 2$).

Additionally, similar to Fig. 2, C_{DP} and C_{BP} tended to maintain G_{sd} better than C_D and C_B. This result suggests that by using the modified D_{ij} obtained from applying the *Physarum* algorithm, it is possible to identify important nodes for maintaining G_{sd}. However, since the links with larger D_{ij} vary depending on the set total flow I_0. It is necessary to investigate the I_0 dependency of each centrality to determine which nodes can maintain G_{sd} for any given I_0.

3.4 Dependence of Centrality on Total Flow

We investigated the total flow I_0 dependency of each centrality to determine whether nodes can maintain G_{sd} for any I_0. Specifically, simulations similar to those in Fig. 3 of Sect. 3.3 were conducted with varying I_0. The total flow values tested were $I_0 = 0.01, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 7, 10$.

To compare how well each centrality maintains G_{sd} when setting I_0, we calculated $\langle G_{sd} \rangle$ for each centrality. $\langle G_{sd} \rangle$ is the expectation value, derived from the area under the curve representing the relationship between node removal rate and G_{sd}, as shown in Figs. 2 and 3. The results are shown in Fig. 4. The horizontal axis shows the total flow I_0, and the vertical axis shows the expectation value $\langle G_{sd} \rangle$.

Figure 4 shows that C_P consistently achieves high $\langle G_{sd} \rangle$ across all I_0 values, highlighting its superior performance compared to other centralities. Both C_P, C_{DP}, and C_{BP} peak around $I_0 = 2$, with C_{DP} showing particularly significant fluctuations. Specifically, C_{DP} shows a significant decrease in $\langle G_{sd} \rangle$ with low total flow such as $I_0 = 0.01, 0.05, 1$, as well as when I_0 exceeds 2. In contrast, C_D and C_B, which do not utilize the *Physarum* algorithm, show no change in performance with varying I_0.

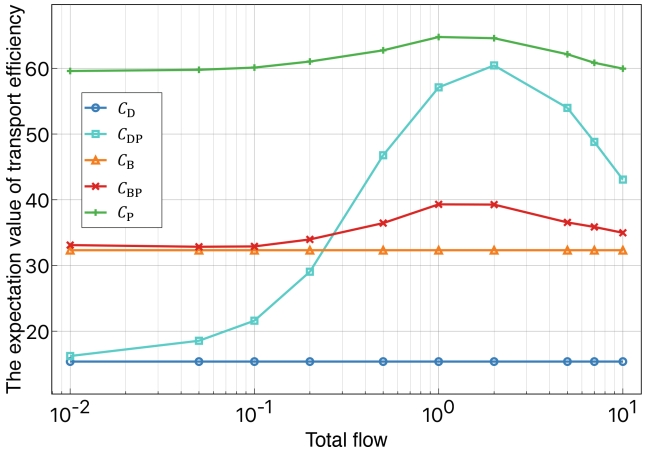


Fig. 4. The expectation value $\langle G_{sd} \rangle$ with respect to the total flow I_0 (Number of nodes: 100, average of 20 network patterns)

The variation in $\langle G_{sd} \rangle$ for C_{DP} and C_{BP} with total flow can be attributed to differences in link conductivity caused by the *Physarum* algorithm. When an appropriate I_0 is set, the links on the shortest path between s and d have higher D_{ij} values, while the links at the ends that are not traversed have lower D_{ij} values. The application of the *Physarum* algorithm enables the link conductivity D_{ij} to reflect the importance related to path transport. Therefore, even centralities that handle only local information, such as degree centrality, can identify nodes crucial to path transport by using D_{ij} at a given appropriate I_0.

On the other hand, setting an I_0 where the entire network is uniformly thick or thin results in performance similar to the case where $D_{ij} = 1$. For instance, as shown in Fig. 1(b) and the left part of Fig. 4, extremely low I_0 causes insufficient transportation throughout the network, leading to a uniform D_{ij} with no significant variation. Conversely, as shown in Fig. 1(c) and the right part of Fig. 4, extremely high I_0 spreads transportation to end links that are not traversed, resulting in a uniform D_{ij}. In these cases, since there is no difference from the situation where $D_{ij} = 1$, C_D and C_{DP}, as well as C_B and C_{BP}, end up selecting similar nodes.

Based on the above, even when considering variations in total flow, C_P evaluates the contribution of nodes to maintaining transportation paths.

4 Conclusions

In this study, we investigated the effectiveness of *Physarum* centrality (C_P) as a metric for evaluating each node's contribution to stable transportation within network paths. We applied the *Physarum* algorithm to Unit Disk Graphs and compared its performance with weighted degree and betweenness centrality. The conductance G_{sd} between the source node s and destination node d was used to

evaluate transport efficiency. The results showed that C_{P} maintained a higher G_{sd} compared to weighted degree and betweenness centrality, regardless of the node removal rate. Even varying total flow I_0, C_{P} outperformed other centralities. These findings indicate that C_{P} effectively ensures stable transportation while minimizing the number of intermediate nodes, regardless of total flow.

References

1. Zhang, Y., Zhang, Z., Wei, D., Deng, Y.: Centrality measure in weighted networks based on an amoeboid algorithm. J. Inform. Comput. Sci. **9**(2), 369–376 (2012)
2. Tero, A., Kobayashi, R., Nakagaki, T.: *Physarum* solver: a biologically inspired method of road-network navigation. Phys. A **363**(1), 115–119 (2006)
3. Watanabe, S., Tero, A., Takamatsu, A., Nakagaki, T.: Traffic optimization in railroad networks using an algorithm mimicking an amoeba-like organism, *Physarum* plasmodium. Biosystems **105**(3), 225–232 (2011)
4. Houbraken, M., Demeyer, S., Staessens, D., Audenaert, P., Colle, D., Pickavet, M.: Fault tolerant network design inspired by Physarum polycephalum. Nat. Comput. **12**, 277–289 (2013)
5. Freeman, L.C., Borgatti, S.P., White, D.R.: Centrality in valued graphs: a measure of betweenness based on network flow. Social Netw. **13**(2), 141–154 (1991)
6. Newman, M.E.J.: A measure of betweenness centrality based on random walks. Social Netw. **27**, 39–54 (2005)
7. Liu, Y., Deng, Y., Jusup, M., Wang, Z.: A biologically inspired immunization strategy for network epidemiology. J. Theor. Biol. **400**, 92–102 (2016)
8. Kwon, H., Choi, Y.H., Lee, J.M.: A *Physarum* centrality measure of the human brain network. Sci. Rep. **9**(1), 5907 (2019)
9. Tanizawa, T., Havlin, S., H.E, Stanley.:Robustness of onionlike correlated networks against targeted attacks. Phys. Rev. E, **85**(4), 046109 (2012)

Resolving the Social Dilemma in Internet Routing Security Adoption Through Altruism, Bounded Rationality, and Network Externality

Keisuke Ishibashi(✉) and Tatsuya Oryu

International Christian University, 3-10-2, Osawa, Mitaka, Tokyo 181-8585, Japan
ikeisuke@icu.ac.jp

Abstract. Some types of Internet security countermeasures require cooperative adoption to effectively defend against cyberattacks, such as authentication mechanisms that prevent spoofing attacks. In these mechanisms, both the information source and the receiver must participate: the source signs the information, and the receiver verifies it. Adoption by only one party is insufficient, creating a social dilemma where cooperative adoption maximizes social welfare, but individual incentives diminish if others do not adopt. Resource Public Key Infrastructure (RPKI) is an example of such a mechanism, protecting against spoofing in Internet routing. Its effectiveness depends on the adoption rate across network paths and has been hindered by a low adoption ratio.

However, recent trends show a moderate increase in RPKI adoption. This paper investigates the reasons behind this adoption, considering the roles of altruism, bounded rationality, and the influence of network externality. Through multi-agent simulations, we explore how these factors interact within different network structures, including the impact of central nodes with high betweenness centrality on the propagation of adoption. Our findings suggest that while altruism and bounded rationality are crucial for overcoming the social dilemma, the underlying network externality also play a significant role in facilitating spontaneous adoption across the network. These insights have implications for both network operators and policymakers in designing strategies to promote the widespread adoption of security technologies.

Keywords: Social Dilemma · Network Effect · RPKI

1 Introduction

As cyberthreats on the Internet have become critical, various types of countermeasures have been investigated. However, some of them fail to be widely adopted on the Internet not due to technological flaws, but because of social

T. Oryu—He has already graduated from the institute.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 196–208, 2025.
https://doi.org/10.1007/978-3-031-82439-5_17

dilemma, discrepancy of incentives. The Resource Public Key Infrastructure (RPKI) is a countermeasure designed to secure Internet routing [1], which has the adoption issue.

The Internet is composed of approximately 75,000 autonomous systems (ASes), which independently operate their networks and connect with other ASes. Each AS announces its own IP address ranges (known as route information) to other connected ASes using the Border Gateway Protocol (BGP). These announcements are propagated further to other ASes, and eventually, all ASes on the Internet recognize that the AS owns the IP address ranges. They send packets, whose destination addresses fall within those ranges, to that AS, this is how Internet routing achieved in a decentralized manner. During this propagation, if a malicious AS announces spoofed route information - an IP address range it does not own - receiving ASes can send packets with those destination addresses to the malicious AS under certain conditions, leading to packet drops, eavesdropping, or data falsification. This attack, known as BGP hijacking, is a significant Internet threat because it does not exploit a vulnerability in specific software implementations, but rather in the Internet control system itself, making it difficult to mitigate [2].

RPKI is developed to prevent BGP hijacking [1]. It consists of two components: ROA (route origin authorization), which involves the registration of IP address ranges by the source AS of the route announcement, and ROV (route origin validation), which involves the validation of ROA information by the receiver AS of the announcement. For RPKI to work effectively, both ROA and ROV must be adopted. If no AS adopts ROV, then the registration of route information (ROA) becomes meaningless, and similarly, if no AS registers its route information (ROA), checking the received route (ROV) is ineffective. Conversely, adopting ROA (or ROV) benefits other ASes that have adopted ROV (or ROA), highlighting the presence of network externalities. Additionally, when an AS adopts ROV, downstream ASes in the propagation of the route benefit from protection against route hijacking. The adoption of ROV by an AS with high betweenness centrality amplifies these network externalities. These strong network externalities create a social dilemma, where cooperative adoption maximizes social welfare, but individual incentives are insufficient if others do not adopt. In other words, if each AS acts "rationally", neither ROA nor ROV would be adopted, rendering RPKI ineffective. In this context, "rationally" refers to a decision-making process in which each player maximizes its own profit (benefit minus cost) in a selfish way. As a result of the issue, the adoption ratio of RPKI remained low, say, below 10 % before 2019 for ROA [3]. When the adoption ratio of ROV is low, then an AS has little incentive to adopt ROA, and vice versa.

However, although the adoption rate is not yet sufficiently high, a certain number of ASes have adopted ROA and ROV. Furthermore, since 2019, the ROA adoption rate has gradually increased and is expected to reach 50 % by 2024. ROV adoption is not yet widespread, but several studies estimate it be around 23 % [5][6]. The Tier 1 ISPs, which sit at the top of the Internet AS hierarchy, have also started to adopt ROV [7]. In this paper, to address the question

of what mechanisms lead to the spontaneous adoption of such security technologies, we explore the roles of bounded rationality and altruism among networked agents in solving the social dilemma [8]. The concept of bounded rationality has gained attention in behavioral economics, where individuals do not always behave rationally, but deviate due to various factors, such as limited information or random behavior. The concept of altruism stems from evolutionary game theory, where players do not act rationally or selfishly to maximize only their own benefit but consider the welfare of others or their neighboring entities. If we assume that AS behavior exhibits these characteristics and deviates from perfect rationality, there may be a possibility of spontaneous adoption.

To verify this assumption, we model the benefits and costs of ROA and ROV adoption and conduct multi-agent simulations using both random network and Internet-like network, varying the degrees of bounded rationality and altruism. We found that certain degrees of these factors lead to the spontaneous adoption of RPKI. To the best of our knowledge, this paper is the first to confirm the possibility of overcoming the social dilemma in the adoption of network security protocols in complex networks by incorporating agent characteristics from behavioral economics and evolutionary game theory.

The remainder of the paper is organized as follows. In Sect. 2, we briefly review the basics of Internet routing, BGP hijacking, and RPKI. In Sect. 3, we model the benefits and costs of RPKI adoption in terms of network effect. In Sect. 4, we evaluate the possibility of spontaneous adoption of RPKI using multi-agent simulations in various networks, and discuss the implications of our findings. In Sect. 5, we briefly review the related work and conclude our study in Sect. 6.

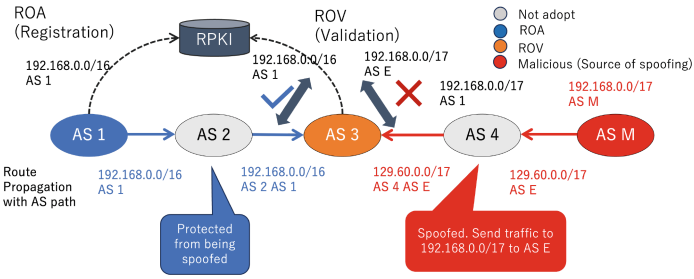


Fig. 1. BGP and RPKI. If AS 3 does not adopt ROV, then it will select the AS path from AS1 or the AS E that corresponds to the width of the IP address range, the AS path length and other policies. In this case, route from AS E is smaller ("/17" indicate that route include 2^{17} addresses) and it might be selected.

2 Internet Routing and RPKI

In this section, we explain how Internet routing works, its vulnerabilities to attacks, and how these attacks can be mitigated.

2.1 Internet Routing and BGP Hijacking

Each AS announces its own IP address range (called a route) to its neighboring ASes, and these announcements are then propagated across the Internet. For this propagation, neighboring ASes exchange their routes using BGP. Route information is propagated with the AS path, a series of ASes leading back to the origin AS of the route. If the same route is announced from different AS with different AS paths, then the receiving AS selects the preferred route based on its policy, such as choosing the shorter path or more specific (narrower) address range, and propagates only selected one as the best route (Fig. 1).

During this route propagation, if a malicious AS announces a spoofed route that is not owned by the AS, the receiving AS might, according to its policy, select this route and further propagate it to neighboring ASes. This type of cyberattack is called BGP hijacking and has become a significant Internet security threat [2].

2.2 RPKI to Prevent BGP Hijacking

To prevent BGP hijacking, several security protocols have been proposed. RPKI is currently mostly promising solution due to its simplicity. In RPKI, owners (ASes) of IP address ranges register certificates that prove ownership of these ranges. These certificates called ROAs, and this process is referred to as adopting ROA hereafter. When an AS receives a route certified by an ROA, it can verify whether the route originates from the legitimate origin AS using the ROA certificate. If there is a mismatch, the receiving AS rejects the route and does not propagate it to neighboring ASes - this process is called ROV.

By implementing both actions, BGP hijacking can be prevented (Fig. 2), however, adopting only one of the two is insufficient for protection. In other words, adopting ROA or ROV has a network effect as these actions increases the benefit for other participants. Additionally, when an AS adopts ROV, it protects downstream ASes in the AS path by rejecting and preventing propagation of spoofed routes. These network effects are numerically modeled in Sect. 3.

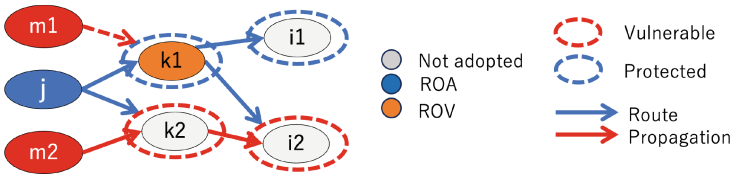


Fig. 2. AS $i2$ is vulnerable under the attack from $e2$, while $i1$ is protected from $e1$ even though it does not adopt ROV, because intermediate AS $k1$ between $e1$ and $i1$ adopts ROV (Assume that route from evil ASes are always preferred due to its range).

3 Modeling RPKI Adoption

In this section, we model the benefits and costs of RPKI adoption, as well as AS behavior, based on the cost-benefit model.

3.1 Benefit and Cost of RPKI Adoption

First, we prepare several notations. Here, each AS is indexed by $i \in \{1, \ldots, n\}(= N)$, where n is the total number of ASes, N refers the set of all AS indexes. Each AS makes its decision on whether it adopts ROA/ROV or not at each time step, where time is discrete (e.g. monthly) and indexed by t.

- A_t: set of ASes that adopt ROA at time t
- V_t: set of ASes that adopt ROV at time t
- $BA_t(i)$: Benefit of AS i that adopts ROA at time t but does not adopt at time $t-1$.
- $CA(i)$: Cost of adopting ROA for AS i.
- $BV_t(i)$: Benefit of AS i that adopts ROV at time t but does not adopt at time $t-1$.
- $CV(i)$: Cost of adopting ROV for AS i.
- $T(i,j)$: Traffic volume from AS i to j [bps]
- $P(i,j)$: set of ASes consisting of the path from AS i to j, including i and j.

In the following, we model $BA_t(i), BV_t(i), CA_t(i), CV_t(i)$. The benefit of adopting ROA or ROV is the reduction of the risk of route hijacking and the risk is the product of the probability of hijacking and its damage.

We assume that the damage is proportional to the volume of hijacked traffic, represented by $\alpha T(i,j)$ for both AS i and j, where α is the proportional coefficient (i.e. dollars per bps). In addition, we assume that the probability of hijacking is the ratio of number of (possibly malicious) ASes that can successfully inject spoofed routes for traffic AS i to AS j. The conditions that traffic from AS i to AS j can be hijacked from an evil AS e at time $t-1$ are as follows (Fig. 2):

- AS j does not adopt ROA, or
- AS j adopts ROA but all ASes in the AS path $P(e,i)$ do not adopt ROV[1].

Remind that the direction of traffic and route propagation is inverse, and for traffic from AS i to AS j, the route origin is AS j and receiver (possibly spoofed) AS is i.

Thus the risk, product of damage and probability, of traffic from AS i to AS j being hijacked at time $t-1$ is

$$\alpha T(i,j), \quad \text{if } j \notin A_{t-1} \tag{1}$$

$$\alpha T(i,j) \frac{\#\{e \in N \setminus \{i,j\} \mid P(e,i) \cap V_{t-1} = \emptyset\}}{n-2}, \text{ otherwise,} \tag{2}$$

[1] As described, whether or not spoofed route is selected depends on various conditions. In this paper, we assume that spoofed route is announced with narrower address range and is always selected.

where $\#\{\cdot\}$ indicates the cardinality of the set. Therefore, benefit of AS j adopt ROA at time t is obtained by subtracting (2) from (1) and summing up all i other than j as follows:

$$BA_t(j) = \alpha \sum_{i \in N \setminus \{j\}} T(i,j) \frac{\#\{e \in N \setminus \{i,j\} \mid P(e,i) \cap V_{t-1} \neq \emptyset\}}{n-2} \tag{3}$$

We then model $BV_t(i)$. Compared to ROA, even if AS i does not adopt ROV, it benefits from other AS in the path adopting ROV as (2) and if AS i adopts ROV, the condition $P(e,i) \cap V_{t-1} \neq \emptyset$ always satisfied. Therefore, $BV_t(i)$ is the sum of the traffic to AS i adopting ROA time the ratio of ASes where the path to AS i is not protected by ROV as follows:

$$BV_t(i) = \alpha \sum_{j \in N \setminus \{i\} \cap A_{t-1}} T(i,j) \frac{\#\{e \in N \setminus \{i,j\} \mid P(e,i) \cap V_{t-1} = \emptyset\}}{n-2}. \tag{4}$$

The costs for ROA and ROV adoption are primarily related to installation cost and the risk of misoperation. The former refers to manual configuration or setting up servers, while the latter involves communication failure if a route is incorrectly rejected. These barriers are discussed [12]. It is difficult to quantitatively estimate, but we assume that those are proportional to the logarithm of AS size as

$$CA_(i) = \beta_A \log(\deg(i)) \tag{5}$$

$$CV_(i) = \beta_V \log(\deg(i)), \tag{6}$$

where we approximate the AS size with its degree (number of peering AS) $\deg(i)$.

3.2 AS Behavior Model Assuming Rationality

Based on the above cost-benefit model, AS behavior at each time step t is modeled as follows:

- An AS i that did not adopt ROV at time $t-1$ will adopt ROV if $BV_t(i) > CV_t(i)$.
- An AS i that did not adopt ROA at time $t-1$ will adopt ROA if $BA_t(i) > CA_t(i)$.

In this paper, we assume that once an AS adopt ROA or ROV, then it will discontinue them, as running costs are negligible. (In practice, the abandonment of ROV/ROA has not been observed).

3.3 AS Behavior Model Incorporating Bounded Rationality and Altruism

It is unlikely that each AS strictly calculates its benefits and costs according to Eqs. (3)(4)(5)(6). Here, we incorporate bounded rationality and altruism into

the model. Under bounded rationality, we assume that each AS does not strictly decide adoption based on the positive or negative of profit (benefit minus cost), but instead makes probabilistic decisions based on the profit. We adopt a logit model [8] for ROV as follows:

$$\Pr\left[\text{AS } i \text{ adopt ROV at time } t\right] = \frac{1}{1 - \exp\left(-\lambda(BV_t(i) - CV_t(i)\right)}, \tag{7}$$

where λ reflects the uncertainty of an AS in calculating the profit. $\lambda = 0$ implies that an AS adopts ROV with a 50 % probability at each time step, ignoring cost and benefit, which would be highly irrational. Bounded rationality in adopting ROA is defined in the same way.

To model altruism, we extend $BV_t(i)$ to $BV_t(i,k)$, which represents the benefit of AS k from the adoption of ROV by AS i at time t. $BV_t(i,k)$ is calculated by counting the malicious AS e whose path to k is not ROV-protected at time $t-1$ but includes i, meaning that i can protect k by adopting ROV, as follows:

$$\begin{aligned} &BV_t(i,k) \\ =&\alpha \sum_{j \in N \setminus \{i,k\} \cap A_{t-1}} T(i,j) \frac{\#\{e \in N \setminus \{i,k,j\} \mid P(e,k) \cap V_{t-1} = \emptyset, i \in P(e,k)\}}{n-3}. \end{aligned} \tag{8}$$

We introduce two types of altruism, global and local. In the former, each AS considers not only its own profit, but also profits of all n ASes, while in the latter, the AS considers the profits of its neighboring ASes.

$$BV_t^{ga}(i) = BV_t(i) + \mu \sum_{k \neq i} BV_t(i,k) \tag{9}$$

$$BV_t^{la}(i) = BV_t(i) + \mu \sum_{k:k \text{ connect with } i} BV_t(i,k), \tag{10}$$

where μ indicates the degree of altruism for an AS. $\mu = 1$ means that the AS weights the benefits of other ASes (all ASes for global altruism, neighboring ASes for local altruism) equally to its own, representing perfect altruism.

4 Multi-agent Simulation and Results

Using the model prepared in the previous section, we build a multi-agent simulation where each AS decides whether to adopt or not ROV and ROA by comparing their benefit and cost.

4.1 Simulation Configuration

Due to computational limitations, we are unable to simulate a real Internet-scale network. Instead, we generate a pseudo-Internet topology using NetworkX [9] of $n = 100$.

For the traffic matrix $T(i,j)$, we use a gravity model, where the traffic flow from AS i to AS j is proportional to the product of the total traffic from AS i and total traffic to AS j, as

$$T(i,j) \propto \left(\sum_k T(i,k)\right)\left(\sum_k T(k,j)\right). \tag{11}$$

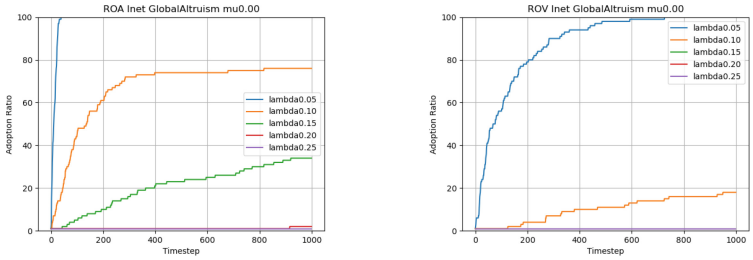
Furthermore, we assume that both $\sum_k T(i,k)$ and $\sum_k T(k,j)$ are proportional to the size of the AS, which is approximated by the square root of the degree of the AS. Thus, $T(i,j)$ can be approximated as

$$T(i,j) = t\sqrt{\deg(i))\deg(j)}, \tag{12}$$

where t is the proportionality coefficient.

We introduce several parameters into the model. Since AS behavior is determined by the difference between cost and benefit, these parameters can be combined to the following cost ratio r:

$$r = \frac{t\alpha}{\beta_V} = \frac{t\alpha}{\beta_A}. \tag{13}$$



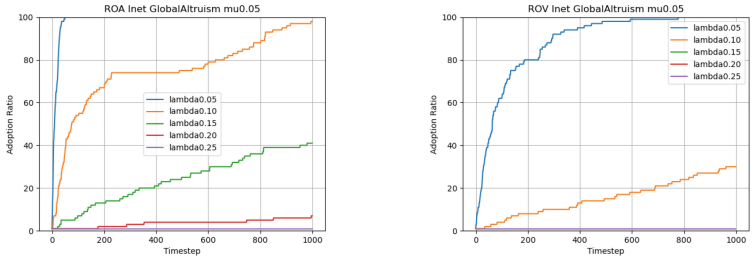
(a) ROA (b) ROV

Fig. 3. Adoption Evolution with Bounded Rationality Only

4.2 Results

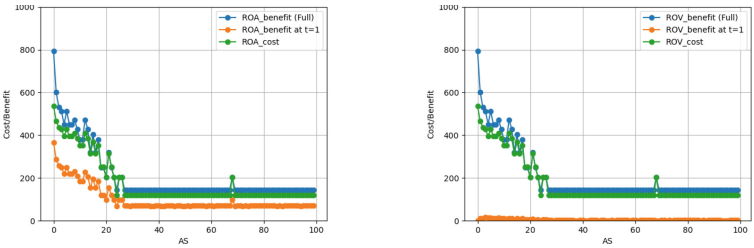
Bounded Rationality and Altruism. ASes are not expected to behave perfectly rationally; instead, they deviate from rationality. Here, we evaluate the effect of this deviation using the model introduced in Sect. 3.3, incorporating bounded rationality and altruism.

We examine the evolution of ROA and ROV adoption, varying λ, the degree of bounded rationality defined in (7), while keeping μ, the degree of altruism defined in (10), fixed. In the following, we fix the cost ratio as $r = 120$. Figures 3 and 4 show the evolution with $\mu = 0$ (no altruism, bounded rationality only) and



(a) ROA (b) ROV

Fig. 4. Adoption Evolution with Bounded Rationality and Global Altruism

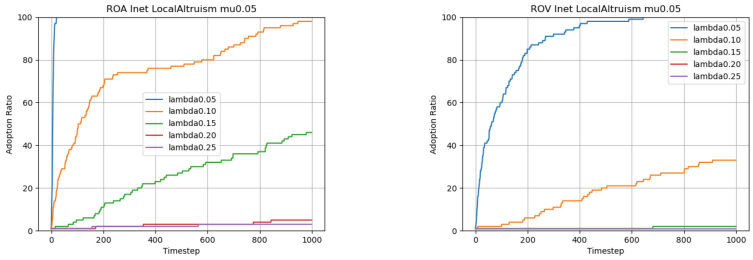


(a) ROA (b) ROV

Fig. 5. Cost and benefit comparison for each AS. ASes are indexed in descending order of their betweenness centralities.

$\mu = 0.05$ (both global altruism and bounded rationality), respectively. As shown, with moderate bounded rationality ($\lambda \leq 0.1$), both ROA and ROV are spontaneously adopted. Altruism also contributes to adoption, though its effect does not seem indispensable. Figure 6 shows the case for local altruism. Compared to Fig. 4, there is little reduction in adoption, and local altruism is sufficient to drive adoption. Figure 7 demonstrates the detailed evolution of adoption in the case of local altruism, with $\lambda = 0.05$ and $\mu = 0.1$. We can observe that the adoption begins with the core ASes, which subsequently leads to the adoption of ROA by other ASes. In turn, this prompts the adoption of ROV by additional ASes. Figure 5 shows the breakdown of costs and benefits for each AS at the time of its first adoption of ROV/ROA (labeled as "$t = 1$"). In both cases, ROA and ROV, the costs for all ASes exceed their benefits initially. Even in this scenario, with bounded rationality, there is still a chance for an AS to adopt ROA or ROV. Interestingly, when considering the cost ratio parameter, once all ASes adopt ROA and ROV, their benefits exceed the costs (labeled as "full"). Therefore, although in the short term, the benefits do not exceed the costs, the results indicate that, in the long term, they can.

For comparison, we simulate the case for a random graph with bounded rationality and global altruism (Fig. 8). This topological change results in a significant



(a) ROA (b) ROV

Fig. 6. Adoption Evolution with Bounded Rationality and Local Altruism

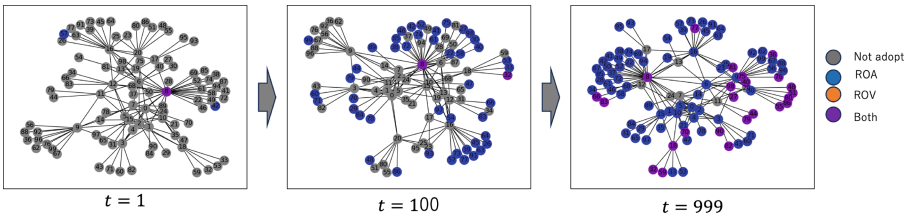


Fig. 7. An example of RPKI adoption evolution. Adoption starts at the core of the network and propagates to its fringe.

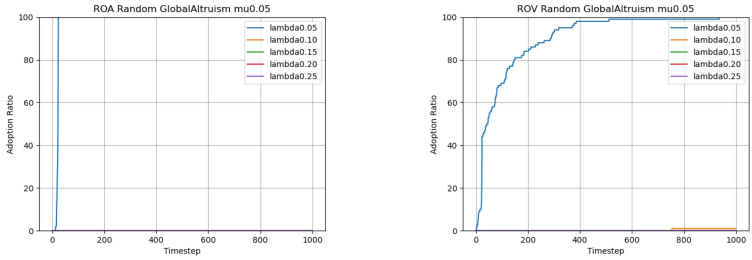
reduction in adoption, which highlights that a small number of high-centrality nodes contribute to both early and sustained adoption of RPKI.

The simulation results can be summarized the follows:

- If all ASes act rationally, no AS can be expected to spontaneously adopt RPKI. Therefore, it is required to enforce or provide external incentive for initial adoption. However, a small number of AS adoption is sufficient, provided that the cost ratio is moderate, the topology is Internet-like, and initial ASes are selected from core ASes with high betweenness.
- While the benefit of adopting RPKI is zero when no AS adopts, the potential benefit of the core ASes is high. There is an opportunity to bridge this gap through bounded rationality or local altruism, particularly for the core ASes with large number of neighbors. After initial adoption, other ASes begins to gain benefits, and adoption may then progress spontaneously.

5 Related Works

The status of RPKI adoption has been well surveyed for ROA in terms of the coverage ratio of ROA-protected IP address ranges [3] [4]. Although ROV adoption cannot be measured directly as ROA, several studies have proposed methods to infer the adoption rate [5][10][11][12], confirming a gradual increase in ROV adoption.



(a) ROA (b) ROV

Fig. 8. Adoption Evolution with Bounded Rationality and Global Altruism in Random Network

Even partial adoption of ROV is beneficial in mitigating BGP hijacking. Specific adoption patterns or thresholds necessary for effective protection have also been explored [10][13]. Based on these investigations, Gilad et al. have presented several proposals to accelerate RPKI adoption [14], addressing the incentive issue often framed as a "chicken and egg" problem.

In our previous work, we quantitatively modeled incentives for rational behavior among ASes [15]. However, that work could not fully explain the current spontaneous and gradual adoption of ROV. In this paper, we relax the assumption of complete rationality and instead consider bounded rationality and altruism as potential drivers for spontaneous adoption.

6 Conclusion

In this paper, we seek the possibility of overcoming the social dilemma through bounded rationality and altruism. The results from the multi-agent simulation suggest that some degree of bounded rationality and altruism (specifically local altruism) are sufficient to solve the social dilemma and achieve the benefit of security protocol, even when the benefits appear smaller than the costs from a short-term perspective. On the academic side, this study makes a contribution to the inter-disciplinary fields of network science, network security and behavioral economics.

From a practical standpoint, the implication of these results extend to network operators, protocol designers, and policy makers. For network operators, especially large Internet Service Providers (ISPs) and core ASes, the findings suggest that by taking the lead in adopting security protocols like RPKI, they can create a large network effect that encourages other ASes to follow, which in return benefits the core ASes themselves. Protocol designers can better understand which types of network security protocols are likely to be adopted spontaneously based on network topology and the presence of bounded rationality or altruism. Although the degree of altruism or bounded rationality cannot be

controlled, but can be estimated through the adoption progress of other protocols, thus, policy makers can gain insight to craft policies that either reduce cost of adoption or increase its benefit (e.g. through subsidies or tax incentives) given the estimated altruism and bounded rationality. They can also enforce adoption through regulation, particularly core ASes, thus ensuring the security of the Internet routing.

Since some of the results may depend on the network size, further evaluation with Internet-scale networks remains a task for future work. Additionally, investigating the reasons behind the regional difference reported in [12], by controlling parameters of our model, will provide further valuable insights.

Acknowledgement. This work was supported by the JSPS KAKENHI grant number JSPS22K11999.

References

1. Lepinski, M., Kent, S.: An infrastructure to support secure internet routing.: RFC 6480. Internet Engineering Task Force (2012)
2. The Whitehouse: Roadmap to enhancing internet routing security: September 2024 (2024). https://www.whitehouse.gov/wp-content/uploads/2024/09/Roadmap-to-Enhancing-Internet-Routing-Security.pdf
3. NIST RPKI Monitor. https://rpki-monitor.antd.nist.gov/
4. Wahlisch, M, Schmidt, R. Schmidt, T.C., Maennel, O., Uhlig, S., Tyson, G.: RiPKI: the tragic story of RPKI deployment in the web ecosystem. In: ACM HotNets (2015)
5. Chung, T, et al.: RPKI is coming of age: a longitudinal study of RPKI deployment and invalid route origins. In: Internet Measurement Conference (2019)
6. Hlavacek, T., et al.: Keep your friends close, but your Routeservers closer: insights into RPKI validation in the internet. In: 32nd USENIX Security Symposium (2023)
7. NTT improves security of the internet with RPKI origin validation deployment (2020). https://www.gin.ntt.net/ntt-improves-security-of-the-internet-with-rpki-origin-validation-deployment/
8. Carpenter, J., Robbett, A.: Game Theory and Behavior. The MIT Press (2022)
9. Software for Complex Network. https://networkx.org/documentation/stable/index.html
10. Gilad, Y., Cohen, A., Herzberg, A., Schapira, M., Haya Shulman, H.: Are we there yet? On RPKI's deployment and security. In: Network and Distributed System Security (NDSS) Symposium (2017)
11. Li, W., Lin, Z., Ashiq, M.I., Aben, E., Fontugne, R., Phokeer, A., Chung T.: RoVista: measuring and analyzing the route origin validation (ROV) in RPKI. In: IMC'23. Montreal, Canada (2023)
12. Testart, C., Wolff, J., Gouda, D., Fontugne, R.: Identifying current barriers in RPKI adoption. In: TPRC 2024 The Research Conference on Communications, Information & Internet Policy (2024)
13. Ando, M., Okada, M., Kanaoka, A.: Simulation study of BGP origin validation effect against mis-origination with internet topology. In: 2017 12th Asia Joint Conference on Information Security (AsiaJCIS) (2017)

14. Gilad, Y, Hlavacek, T., Herzberg, A. , Schapira, M.: Perfect is the enemy of good: setting realistic goals for BGP security. In: Proceedings of the 17th ACM Workshop on Hot Topics in Networks, pp. 57–63 (2018)
15. Oryu, T, Ishibashi, K.: Simulation evaluation of RPKI deployment based on cost-benefit analysis. In: 2021 International Conference on Emerging Technologies for Communications (ICETC 2021) (2021)

Link Analysis Ranking

Predicting Interactions in the Weapons of Mass Destruction Knowledge Graphs

Abhigya Agrawal(✉), Md Saidul Hoque Anik, and Ariful Azad

Department of Intelligent Systems Engineering, Indiana University, Bloomington, USA
{abagra,mdshoque,azad}@iu.edu

Abstract. In this paper, we apply graph machine learning methods to predict unseen interactions within the Weapons of Mass Destruction (WMD) dataset, developed by DARPA and IARPA.This dataset captures complex online activities, including sales, purchases, and forum discussions, with a focus on topics such as weapons, explosives, and other sensitive subjects. We represent the data as a knowledge graph, where nodes correspond to entities and edges denote relationships between them. Among various knowledge graph embedding techniques and graph neural networks, semantic matching models like DistMult demonstrate the ability to accurately predict 84% of relations, particularly due to their strength in capturing the one-to-many relationships common in the WMD data. To streamline the analysis, we implement an automated pipeline that stores the knowledge graph in a Neo4j database, extracts subgraphs using Cypher queries, trains knowledge graph embedding models on these subgraphs, predicts links, and reintegrates high-confidence edges back into the main graph.

Keywords: Knowledge graph · embedding · online forum

1 Introduction

In today's digital landscape, online forums play an important role in facilitating social and economic transactions [7,12]. Predicting future interactions based on historical forum activities has significant applications in areas such as advertising, recommendation systems, and cybersecurity [20,21]. For instance, security agencies monitor and analyze forum posts, comments, and interactions to detect suspicious activities. In line with these objectives, the U.S. Defense Advanced Research Projects Agency (DARPA) compiled an extensive dataset with online sales, purchases, and forum discussions, including those related to weapons, explosives, and other sensitive topics[1]. This dataset is referred to as the Weapons of Mass Destruction (WMDs) dataset. This paper explores machine learning (ML) algorithms to predict future interactions including potential threats from

[1] https://www.darpa.mil/program/modeling-adversarial-activity.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 211–223, 2025.
https://doi.org/10.1007/978-3-031-82439-5_18

these multi-modal WMD datasets. Security agencies can take proactive measures to prevent harm and save lives by accurately identifying future activities.

One of the key challenges in predicting future activities from the WMD data is its multi-modal and unstructured nature. The data captures entities, including individuals, products, topics, publications, forums, and their interactions. It also includes both text and numerical descriptions with temporal aspects. To address this complexity, we use a knowledge graph (KG) to model the data. A KG organizes information into a network of nodes (representing entities) and edges (representing relationships), with entities and relations having additional features [6]. Modeling the WMD data as a knowledge graph offers two main advantages. First, it provides a structured representation of the data, which is stored in a graph database, allowing for easy retrieval and augmentation with additional information. Second, it enables us to approach future interaction predictions as a link prediction problem, a well-established technique in the literature (Fig. 1).

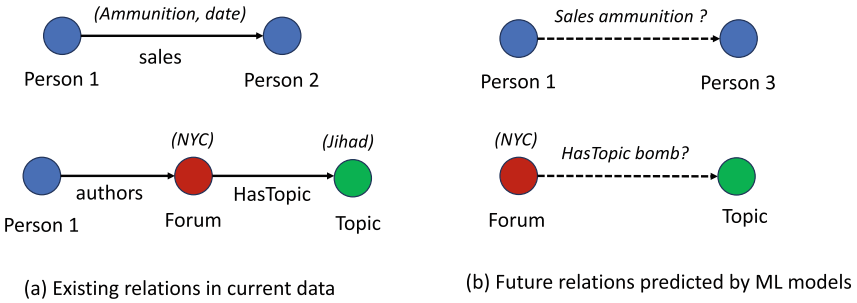


Fig. 1. (a) Examples of sale, author, and HasTopic relations in the WMD data. (b) The top right prediction task might assess whether Person 1 is likely to sell ammunition to Person 3, based on the fact that Person 1 previously sold ammunition to Person 2. Similarly, the bottom right prediction might evaluate whether a forum that discusses topics related to New York City will eventually discuss the topic of bombs. These predictions leverage patterns observed in historical data to anticipate future interactions.

Over the last decade, the link prediction problem has been solved by various graph embedding algorithms [5,13,17], graph neural networks [8,18], and knowledge graph embedding (KGE) models [1,9].Given the extensive research in this area, we do not aim to develop new ML methods for link prediction on the WMD knowledge graph. Instead, we focus on identifying the most suitable models for predicting future transactions and understanding the underlying reasons for their observed performance. To this end, we employed a classic node embedding algorithm, node2vec [5], the Graph Convolutional Network (GCN) [8], and several KGE models. We found that certain classes of KGE models, particularly those capable of capturing one-to-many relations, performed well in predicting new relationships within the WMD knowledge graph. This is likely due to the

prevalence of one-to-many relations in the dataset, such as sale (a person sells multiple products), author (a person authors several topics), etc. In contrast, node2vec and GCN did not perform as well as the top KGE models, primarily because of their limitations in capturing the types of relations in our dataset.

Based on our experiments, this paper makes the following contributions. (1) We utilize a knowledge graph model to represent multi-modal WMD data from DARPA. (2) We compare various supervised and unsupervised embedding algorithms to predict new interactions within the WMD knowledge graph. Based on these experiments, we identified specific KGE models that effectively capture the unique relations in the data.(3) To automate the analysis, we develop a comprehensive pipeline that stores the knowledge graph in a Neo4j database, extracts subgraphs via Cypher queries, trains KGE models on subgraphs, predicts links, and reintegrates high-confidence edges back into the original graph.

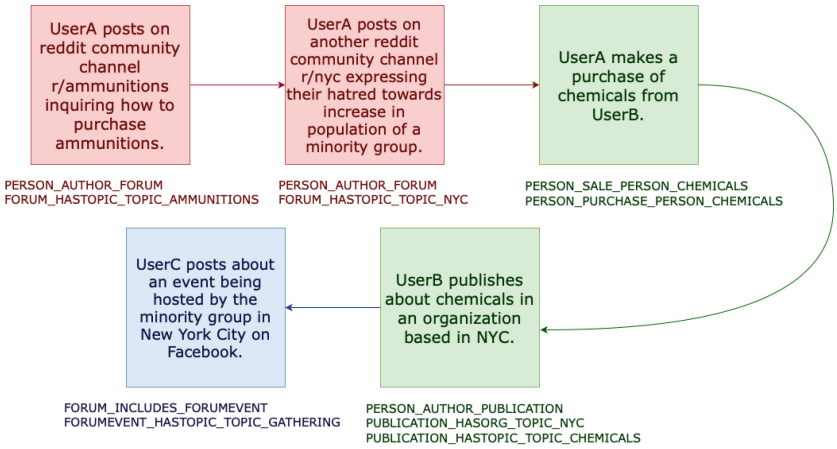


Fig. 2. Examples in WMD dataset and their corresponding triples.

2 Method

2.1 The Weapon of Mass Destruction (WMD) Social Data

The WMD data generated by the Modeling Adversarial Activity (MAA) Program in DARPA is prepared to help develop mathematical and computational techniques for modeling adversarial activity. This multi-modal data captures online sales/purchase activities including weapons, explosives, and fertilizers as well as forum discussions on various topics including bombing, jihad, and specific locations. Figure 2 provides several examples of the data content and illustrates how it can be structured into knowledge graph triples.

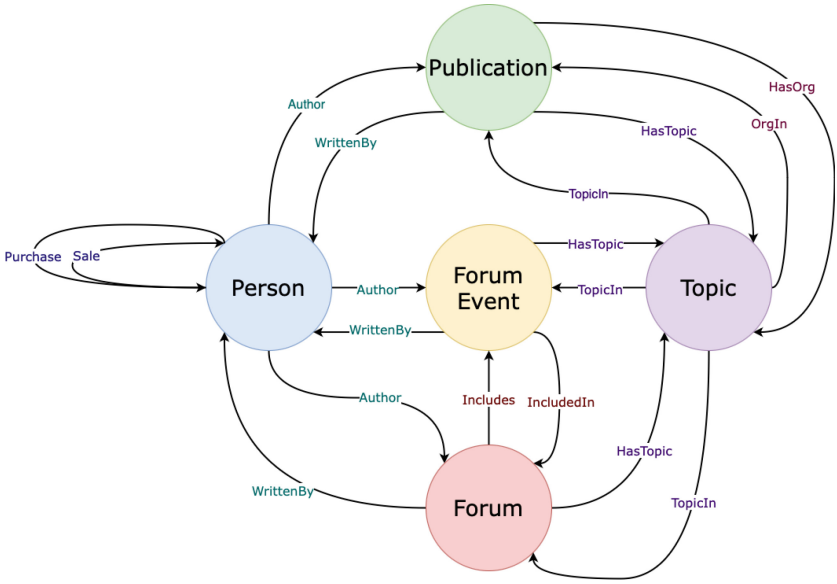


Fig. 3. Visualization of the WMD dataset entities and relations.

2.2 Representing Online Transactions as Knowledge Graphs

A knowledge graph representation of the data is the best way to achieve our goal of predicting forthcoming interactions. WMD data contains five types of entities: Person, Forum Events, Forum, Publication, and Topic. These entities are linked to each other through different types of relations as shown in Fig. 3. A 'Sale' or a 'Purchase' occurs between two persons, and these relations are inverse of one another. and are of the type 1-to-many. A person can be an 'Author' of Publications, Forums, and Forum Events. Publications, Forums, and Forum Events are 'WrittenBy' a Person. 'Author' and 'WrittenBy' relations are also inverse relations. A person can be an author for multiple Publications, Forums, and Events and vice versa. Hence both 'Author' and 'WrittenBy' relations are 1-to-many relations. Publications, Forums, and Forum Events will have certain Topics they discuss and are linked to Topics via the relation 'HasTopic'. 'HasTopic' and 'TopicIn' are inverse to each other and are 1-to-many relations. A Forum Event is 'IncludedIn' a Forum and a Forum 'Includes' a Forum Event. These relations are inverse of each other and one Forum can include multiple Forum Events and one Forum Event can be present in multiple Forums, making them 1-to-many relations. A Publication has an organization in Topic that is represented by the relation 'HasOrg'. 'OrgIn is the inverse of 'HasOrg' relation and they both follow the 1-to-many relation type. The original data file contains five types of relations, their inverse relations are added to the data while pre-processing. These five relations are: 'Sale', 'Author', 'Includes', 'HasTopic', and 'HasOrg'. These relations will be referred to as the main five relations in the following sections.

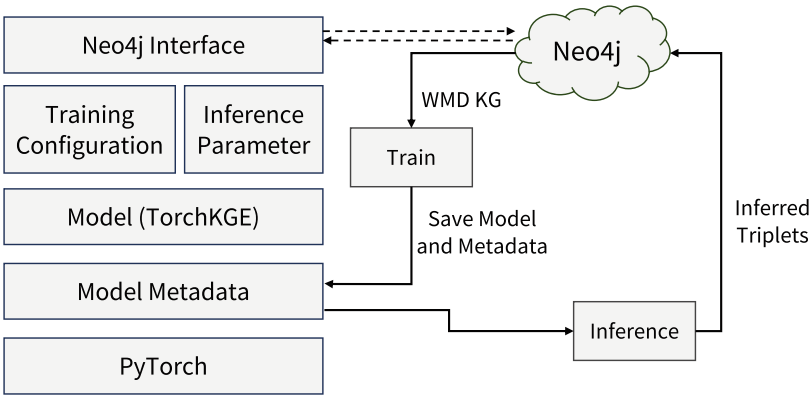


Fig. 4. Software framework on the left and the training and inference pipelines on the right.

2.3 The Overall Software Framework

Fig. 4 shows the software framework used to train knowledge graphs retrieved from the Neo4j database. We rely on the TorchKGE library for the training and testing of KGE models. Our framework has configurable training and inference pipelines. The training component can fetch triplets from the Neo4j database and perform model training to generate embeddings for entities and relations. After the training is finished, it stores the generated embeddings and model metadata locally. The inference component uses these model artifacts to create new facts and assign scores to them. The inferred triplets are stored locally. The user can upload any number of these triplets back to the Neo4j database. We deployed these pipelines on a compute cluster, allowing users to apply them to any knowledge graph for both training and inference purposes.

2.4 Predicting Interactions by KG Embedding

One of the key components of our data analysis pipeline is knowledge graph embedding (KGE). Most KGE models represent the head, relation, and tail of a triple by embedding vectors in Euclidean space such that a combination of head and relation embedding vectors results in the tail embedding vector. The score function of a knowledge graph embedding model measures how distant are two nodes compared to the relation between them. KGE models are trained to reduce the score between the combination of head and relation embedding and tail embedding. These embeddings are evaluated by testing their performance on knowledge graph completion tasks. Knowledge graph completion task includes predicting the unseen relations r between two existing entities: (h,?,t). This task is also known as a link prediction task or interaction prediction task. Since KGE is a well-studied problem in the literature, we do not aim to develop new models. Instead, we compare two classes of KGE models for WMD data.

Table 1. Types of relations supported by different KGE models.

Model	Symmetric	Anti-Symmetric	Inversion	Composition	1-to-N
TransE	No	Yes	Yes	Yes	No
TransH	Yes	Yes	Yes	Yes	Yes
TransR	Yes	Yes	Yes	Yes	Yes
DistMult	Yes	No	No	No	Yes
RESCAL	Yes	Yes	Yes	Yes	Yes
ComplEx	Yes	Yes	Yes	Yes	Yes

The first class of models are called translational models that aim to reduce the Euclidean distance between head + relation embeddings and tail embeddings. Different types of translational models opt for different representations of head, relation, and tail embeddings. TransE [1] represents head, relation, and tail in the same semantic space of $\mathbb{R}^d$. Since both relations and entities are in the same dimensional space, TransE is unable to capture symmetric and 1-to-many relations. TransR [9] proposes a solution for TransE's drawbacks by having two different spaces for entities and relations. The entities in space $\mathbb{R}^k$ are projected to the relation space $\mathbb{R}^d$ by a projection matrix $M_r \in \mathbb{R}^{kxd}$. TransR can capture different kinds of relations at the cost of performance. TransH [19] proposes a method to perform translation on the hyperplane. Every relation has a relation-specific hyperplane represented by w_r. The head and tail vectors h, t are projected to the relation-specific hyperplane by w_r using $h_\perp = h - w_r^\top h w_r$ and $t_\perp = t - w_r^\top t w_r$. These projected head and tail vectors are then connected via the relation-specific translation vector d_r or just r.

The second model class are called semantic matching models that use a bilinear score function. DistMult [22], RESCAL [10] and ComplEx [16] are examples of semantic matching models. RESCAL represents each entity with a vector and each relation as a matrix to capture the latent semantics of a knowledge graph. The score function (h,r,t) is represented by the vectors h and t in R^d and the relation matrix M_r in $\mathbb{R}^{d \times d}$. The score function captures pairwise interaction between all instances of entities h and t through M_r. RESCAL uses tensor decomposition to solve the equation. RESCAL can be computationally complex and expensive as each relation requires $O(d^2)$ parameters. DistMult [22] proposes a solution to reduce the computational complexity of RESCAL by adding a constraint to the relation matrix M_r to be a diagonal matrix. This constraint also adds a limit to DistMult that the model can only capture symmetric and 1-to-many relations efficiently. The ComplEx model aims to perform embedding with lesser parameters while capturing anti-symmetric relations.

Table 1 shows the types of relations each KGE model can effectively capture. Different models excel at modeling various relation types. For example, TransE [1] performs well with anti-symmetric, inversion, and composition relations, while DistMult [22] is effective at modeling symmetric and 1-to-N relations. Our experiment examines these capabilities, evaluating how well each KGE model captures the diverse relationships present in the WMD knowledge graph.

2.5 Predicting Interactions by GNNs

Graph neural networks (GNNs) [15] are deep learning models that are designed to work with graphs. A knowledge graph can be viewed as a heterogeneous graph with different types of entities and relations. Graph neural networks work on the principle of message passing between two neighboring nodes and combining this information received from their neighbors. Graph convolution networks (GCNs) [8] are a popular type of graph neural networks. Similarly to traditional convolution neural networks, GCNs also operate by convoluting or aggregating the information from the neighboring nodes of the target node. GNNs are widely used to analyze data in a graph structure. They are most often used for node classification and link prediction tasks. In this paper, we focus on the link prediction capabilities of GCN and compare its performance with knowledge graph embedding models' link prediction. Additionally, we run experiments with the embeddings generated by knowledge graph embedding models as features for graph neural networks.

3 Results

3.1 Experimental Settings

Datasets. In this paper, we use four versions of the WMD dataset, each generated using a simulator that creates realistic data based on the actual ontology shown in Fig. 3. The simulator was made available by the IARPA AGILE program[2] to mimic real-world scenarios while ensuring data security and privacy. We used the WMD Small dataset to train KGE models and tune their parameters. The WMD Large dataset is almost double the size of the WMD Small dataset in terms of the number of triples. WMD Small Mod dataset is a modified version of the WMD Small dataset with the five main relations: 'Sale', 'Author', 'Includes', 'HasTopic', and 'HasOrg' and an additional 'Purchase' relation. Similarly, the WMD Large Mod dataset is a modified version of the WMD Large dataset. Table 2 summarizes these datasets.

Table 2. Dataset statistics

Dataset	Entities	Relations	Triples
WMD Small	105932	10	344946
WMD Large	189164	10	699828
WMD Small Mod	105932	6	189598
WMD Large Mod	189164	6	384155

[2] https://www.iarpa.gov/research-programs/agile.

KGE Models. We used six KGE models to predict new relations from the WMD data. These models can be categorized into two groups: Translational models and semantic matching models. Translational models include TransE [1], TransR [9], and TransH [19], which aim to represent relationships as translations in the embedding space. On the other hand, semantic matching models such as RESCAL, DistMult, and ComplEx focus on learning entity and relation embeddings based on similarity in the latent space.

Performance Metrics. Knowledge graph embedding models calculate scores for positive and negative triples using the model's scoring function. A positive triple represents a fact that exists in the knowledge graph, while a negative triple represents a hypothetical fact that does not exist in the graph. To generate the negative triples, either the head entity or the tail entity of a given triple is corrupted and a link prediction task is performed. The metrics used to evaluate these results are Hits@k, MRR, and MR. Hits@k denotes the fraction of positive triples that rank in the top-k among their negative triples. A higher Hits@k score is better. Hits@1 score can be equated to accuracy, where accuracy is defined as the number of positive triples predicted. KGE models assign higher scores to positive triples and lower scores to negative ones, thereby distinguishing between valid and invalid relationships within the knowledge graph. Triples can be ranked according to their scores, where positive triples should ideally receive higher scores than negative triples Marginal Rank or MR is an evaluation metric that marginalizes over the rank of all possible positive triples for the (h, r) or (r, t) pair. For a given pair (h, r) and a corresponding set of tails (t) that represent all tails that form positive triples and $(t_)$ that represent all the tails that form negative triples with (h, r) in the dataset, Marginal Rank will be calculated by taking an average of the ranks for all (h, r, t). MRR or Mean Reciprocal Rank is calculated by taking the mean of the reciprocal of the ranks. Ideally, a lower Marginal Rank and a higher Mean Reciprocal Rank is desired.

Experiment Platforms. In this paper, all the knowledge graph embedding models were trained using TorchKGE python library [2]. The graph neural networks were implemented using the PyTorch-geometric python library [4]. Both of these libraries build upon the PyTorch framework [11] that provides capabilities to define, train, test, and evaluate complex deep learning models. One GPU node instance of NVIDIA A100 GPU [3] is used to run all the code.

3.2 Predicating New Relations in the WMD Knowledge Graphs

In this experiment, our goal is to predict missing relationships (edges) between entities (nodes) in the WMD knowledge graph. As discussed in the methods section, we formulate this as a recommendation problem $(h, r, ?)$ where we aim to predict the tail entity given a head h and relation r. Table 3 compares the results from six KGE models coupled with two loss functions.

We used two loss functions for training these models. The margin loss function [1] works by maximizing the distance between positive and negative triples, ensuring that positive triples are scored significantly higher. On the other hand,

the Binary Cross Entropy (BCE) loss [14] outputs a probability, aiming to increase the likelihood of positive triples while reducing the probability of negative ones.

Table 3 shows that these loss functions give drastically different results when used with the same model. DistMult and RESCAL are both semantic matching models and give their best performance when combined with the BCE Loss function. TransE and TransH are both translational models and give their best performance when combined with Margin loss. TransR requires $O(dk)$ parameters per relation, and this data has ten types of relations, making TransR computationally complex. This results in the worst performance among all the models, even reaching a Hits@k score of 0 with BCE loss. Nevertheless, we observed that DistMult combined with BCE Loss gives the best performance for Hits@k and reaches up to a 0.838 score for Hits@10 and a 0.767 score for Hits@1. Hits@1 can be considered equivalent to an accuracy metric. This will be useful in comparing KGE results with those obtained from graph neural networks. DistMult with BCE also outperforms all the other models in both MRR and MR metric with a 0.79 score for MRR and 1995 rank for MR.

Table 3. Knowledge graph embedding model performance comparison on the WMD Small dataset.

Method	Loss	MRR	MR	Hits@1	Hits@10
TransE	Margin	0.336	2604	0.089	0.654
TransE	BCE	0.236	2180	0.196	0.301
DistMult	Margin	0.392	2979	0.331	0.495
DistMult	BCE	0.79	1995	0.767	0.838
RESCAL	Margin	0.345	3159	0.284	0.45
RESCAL	BCE	0.489	2828	0.379	0.674
TransR	Margin	0.133	11967	0.008	0.31
TransR	BCE	0	30933	0	0
TransH	Margin	0.337	2645	0.087	0.66
TransH	BCE	0.229	2205	0.188	0.298
ComplEx	Margin	0.216	3549	0.15	0.361
ComplEx	BCE	0.549	3266	0.552	0.619

3.3 Relation-Wise Performance of KGE Models

To assess the performance of various KGE models, we analyzed their ability to predict different types of relations, as shown in Fig. 5. DistMult outperforms other models for all relations except "Includes," largely because it excels at capturing 1-to-many relations like "Sale," "Author," and "HasTopic," as indicated

in Table 1. RESCAL performs best for the "Includes" relation due to its flexibility in modeling various relation types. Since 1-to-many relations dominate the dataset, models like DistMult and RESCAL perform the best. However, performance may vary depending on the dataset, so testing multiple models is recommended.

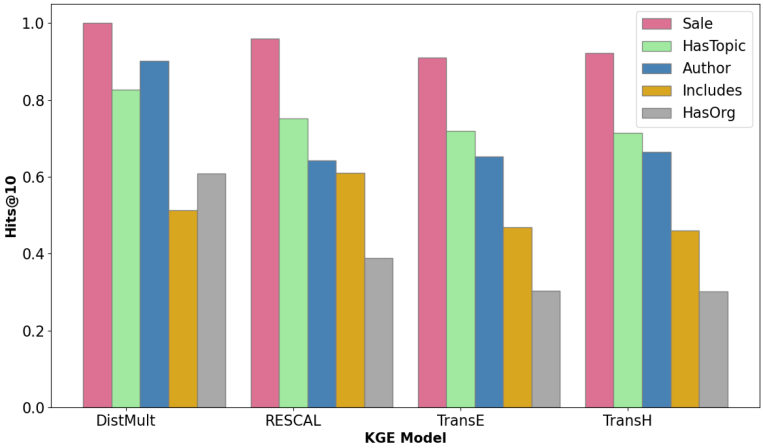


Fig. 5. Relation-wise performance of KGE models.

Table 4. Link Prediction with different methods on WMD Small dataset.

Method	Final Accuracy
DistMult Link Prediction	0.767
GCN Link Prediction	0.7029
Node2Vec + Logistic Regression Link Prediction	0.6884
DistMult Emb + GCN Link Prediction	0.6971

3.4 Comparison of KGE with Node Embedding and GNNs

When a knowledge graph is simplified by ignoring entity and relation types, traditional graph embedding methods and graph neural networks (GNNs) can be used for edge prediction. In our experiment, we applied the node2vec embedding algorithm [5] and trained a logistic regression classifier for link prediction. We also trained a Graph Convolutional Network (GCN) [8] for the same task. Accuracy was used for GCN and logistic regression, while Hits@1 was used for KGE models. As shown in Table 4, the DistMult KGE model outperformed both

node2vec and GCN. Using DistMult embeddings as node features for the GCN did not improve its performance, indicating that KGE models are more effective in capturing entity and relation information that GCN models overlook.

3.5 Scaling and Privacy

The WMD dataset contains sensitive information and ideally, we want to be able to train on a small portion of the dataset and reproduce those results for a larger dataset. To test the scaling capability of KGE models, we ran experiments for WMD Small and WMD Large datasets, as well as for WMD Small mod and WMD Large mod datasets. The results for these experiments are shown in Table 5. The purpose of these experiments is to check if the performance is consistent across different dataset sizes. The DistMult model with BCE loss function is used to train all datasets. The different-sized datasets do show a consistent performance for the Hits@10 metric and Marginal Rank metric. The experiments conclude that to protect the privacy of the data, the knowledge graph embedding models can be trained on a smaller portion of the data and the results can be scaled to a larger dataset size.

Table 5. Knowledge graph embedding model performance for different sizes of data

Dataset	MRR	MR	Hits@1	Hits@10
WMD Small	0.79	1995	0.767	0.838
WMD Large	0.763	2722	0.676	0.811
WMD Small mod	0.255	11907	0.252	0.307
WMD Large mod	0.251	19554	0.211	0.324

4 Conclusions

We developed a software framework for storing and analyzing DARPA's WMD knowledge graph. This framework incorporates various KGE models, graph embedding algorithms, and GNNs to predict new relationships. The best-performing model, DistMult, accurately predicts 84% of relations, outperforming other KGE models, node2vec, and GCNs. DistMult's success is due to its ability to capture the one-to-many relationships prevalent in the WMD data. The analyzed WMD knowledge graph includes typical online activities like sales, purchases, and discussions, offering insights applicable to similar online forum datasets.

Acknowledgements. This research is based upon work supported by the Intelligence Advanced Research Projects Activity (IARPA), through the Advanced Graphical Intelligence Logical Computing Environment (AGILE) research program, under Army Research Office (ARO) contract number W911NF22C0084. This work is also supported by the NSF grant OAC-2339607.

References

1. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (eds.) Advances in Neural Information Processing Systems, vol. 26. Curran Associates, Inc. (2013)
2. Boschin, A.: TorchKGE: knowledge graph embedding in Python and PyTorch. In: International Workshop on Knowledge Graph: Mining Knowledge Graph for Deep Insights (2020)
3. Choquette, J., Gandhi, W., Giroux, O., Stam, N., Krashinsky, R.: NVIDIA A100 tensor core GPU: performance and innovation. IEEE Micro **41**(2), 29–35 (2021)
4. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
5. Grover, A., Leskovec, J.: Node2Vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864 (2016)
6. Hogan, A., et al.: Knowledge graphs. ACM Comput. Surv. (CSUR) **54**(4), 1–37 (2021)
7. Jin, L., Chen, Y., Wang, T., Hui, P., Vasilakos, A.V.: Understanding user behavior in online social networks: a survey. IEEE Commun. Mag. **51**(9), 144–150 (2013)
8. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2017)
9. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 29, no. 1 (2015)
10. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML'11, pp. 809–816. Omnipress, Madison, WI, USA (2011)
11. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library (2019)
12. Pendry, L.F., Salvatore, J.: Individual and social benefits of online discussion forums. Comput. Hum. Behav. **50**, 211–220 (2015)
13. Rahman, M.K., Sujon, M.H., Azad, A.: Force2Vec: parallel force-directed graph embedding. In: 2020 IEEE International Conference on Data Mining (ICDM), pp. 442–451. IEEE (2020)
14. Ruffinelli, D., Broscheit, S., Gemulla, R.: You can teach an old dog new tricks! on training knowledge graph embeddings. In: International Conference on Learning Representations (2020)
15. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE Trans. Neural Netw. **20**(1), 61–80 (2009). https://doi.org/10.1109/TNN.2008.2005605
16. Trouillon, T., Welbl, J., Riedel, S., Gaussier, E., Bouchard, G.: Complex embeddings for simple link prediction. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning, ICML'16, vol. 48, pp. 2071–2080 (2016)
17. Tsitsulin, A., Mottin, D., Karras, P., Müller, E.: VERSE: versatile graph embeddings from similarity measures. In: Proceedings of the 2018 World Wide Web Conference, pp. 539–548 (2018)

18. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
19. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 28, no. 1 (2014)
20. White, R.W., Bailey, P., Chen, L.: Predicting user interests from contextual information. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 363–370 (2009)
21. Wong, G.K., Li, Y.K., Lai, X.: Visualizing the learning patterns of topic-based social interaction in online discussion forums: an exploratory study. Educ. Tech. Res. Dev. **69**(5), 2813–2843 (2021)
22. Yang, B., Tau Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases (2015)

Backbone Extraction in Complex Networks via Embedding-Based Link Prediction

Jason Barbour[1,2](✉), Ali Yassin[3], Hocine Cherifi[4], and Hamida Seba[5]

[1] Graduate Program in Computational Science, American University of Beirut, Beirut, Lebanon
jgb21@mail.aub.edu
[2] Center For Advanced Mathematical Sciences, American University of Beirut, Beirut, Lebanon
[3] Laboratoire d'Informatique de Bourgogne - Université de Bourgogne, Dijon, France
[4] ICB UMR 6303 CNRS - Université de Bourgogne, Dijon, France
[5] Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, 69622 Villeurbanne, France

Abstract. Network backbone extraction is essential for simplifying complex systems while preserving critical structural features. While traditional backbone extraction methods rely on statistical tests and topological attributes, this research presents a novel approach using embedding-based link prediction techniques. We conduct a comparative analysis with classical methods across several real-world networks to assess the effectiveness of different embedding methods. Our findings show that Graph Autoencoders (GAE) and Laplacian Eigenmaps (LEM) based-backbone methods effectively capture regional connections in a hub-and-spoke air transportation network. In contrast, the High Salience Skeleton method excels in preserving overall network connectivity and reachability. Among the methods analyzed, DeepWalk preserves the weight distribution best, and LEM is most effective in maintaining the degree distribution. These results suggest that selecting a backbone extraction method should depend on the analysis's specific network characteristics and objectives. This approach offers valuable insights for transportation, social, and other types of networks.

Keywords: Complex Networks · Backbone Extraction · Link Prediction · Filtering Techniques · Node Embedding

1 Introduction

A backbone is a representation that retains only the most crucial connections of a network. Simplifying a network by extracting its backbone is essential for robust network analysis in various disciplines, including social network analysis [1,2], bioinformatics [3,4], epidemiology [5–7], and transportation systems [8,9]. It can

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 224–236, 2025.
https://doi.org/10.1007/978-3-031-82439-5_19

facilitate significant insight and enhance operational efficiency. However, this process involves a considerable trade-off, as excessive simplification may result in the exclusion of crucial information required for precise network analysis.

Conventional backbone extraction methodologies include statistical and structural filtering techniques [10–15]. Generally, they rely on weight distributions or topological attributes to extract the backbone. This study introduces a novel approach based on link prediction techniques to construct the backbone of a network. Link prediction techniques work by learning patterns in the structure of a network, using similarity measures, probabilistic models, or machine learning to estimate the likelihood of connections between nodes [16]. This probabilistic approach can be used to identify and preserve the most significant edges within the network.

By assessing the probabilities associated with edges, it is possible to differentiate between predictable and unpredictable edges based on the underlying assumptions of the link prediction method. Specifically, our focus lies on the unpredictable edges that deviate from the method's assumptions, akin to approaches used in statistical backbone extraction. This process involves identifying irregularities or anomalies in the network structure, highlighting edges that do not conform to expected patterns, thereby enriching our understanding of the network's intricate dynamics.

The embedding-based link prediction methods can be categorized into three main categories: matrix factorization, random walk-based approaches, and learning-based strategies. Each category offers unique methodologies and advantages for predicting links in networks. To demonstrate the effectiveness of our approach, we choose a method from each category (DeepWalk, Laplacian Eigenmaps, and Modularity-aware Graph Autoencoders). Our comparative analysis includes traditional backbone extraction methods such as the Disparity filter and the High Salience Skeleton [17,18]. We examine the backbones of various real-world networks: the world air transportation network, an American elementary school social network, and the Les Misérables network, focusing on basic topological properties such as the hub-and-spoke topology of the US air transportation network, as well as weights and degree distributions.

Our findings indicate that the proposed methodology not only maintains vital structural attributes, such as node degree and edge weight distribution but also offers a clearer depiction of regional spoke dynamics within the hub-and-spoke structure, a typical feature observed in transportation networks.

The remainder of this paper is structured as follows: Sect. 1 reviews the background on backbone extraction and link prediction methodologies. Section 2 introduces the embedding-based link prediction method for backbone extraction. Section 4 details the experiments. Section 5 presents the results of the comparative analysis. Finally, Sect. 6 offers concluding remarks and outlines directions for future research.

2 Background

2.1 Backbone Extraction

Backbone extraction is used to simplify networks while retaining their key features. Methods can be broadly classified into two categories: statistical and structural [10]. Statistical methods use hypothesis testing or empirical weight distribution techniques to evaluate the significance of edges. The Disparity filter, Polya Urn filter, GloSS, and LANS filter are popular examples of this approach. Hypothesis testing involves comparing observed edge weights against a null model to derive a p-value quantifying the deviation from the null hypothesis. Empirical weight distribution techniques assess edge significance based on the network weight distribution. Structural methods focus on the network's topological attributes to identify a backbone with specific topological properties. Notable examples of this approach include the High Salience Skeleton filter, Metric Backbone filters, and H-Backbone filter [10]. To evaluate the proposed method we concentrate on the most popular techniques derived in these two approaches: The Disparity filter and the High Salience Skeleton.

Disparity Filter [17]**:** It assesses the significance of an edge locally from the viewpoint of the incident nodes considered independently. It assumes that a node u distributes its total strength $s_u = \sum_{z \in \Gamma(u)} w_{(u,z)}$ uniformly among its edges. Based on the defined model, it performs a one-sided right-tail test computing the probability of observing the weight W at least as "extreme" as p_{uv}.

High Salience Skeleton [18]**:** This method operates based on edge salience s_{uv}. Initially, a shortest path tree T_u is constructed for each node u where $T_{uv}(u)$ is 1 if edge (u, v) is part of the shortest path tree rooted at node u, and 0 otherwise. Then, the edge salience is computed following: $s_{uv} = \sum_{u \in N} T_{uv}(u) \times \frac{1}{|N|}$ Where $|N|$ is the total number of nodes. So s_{uv} represents the fraction of shortest path trees that contain edge (u, v). Finally, a threshold is set to remove edges with salience below this threshold.

2.2 Node Embedding

Numerous network embedding methods have been proposed in the literature [19]. These methods can be classified into three main categories: random walk-based, matrix factorization, and deep learning-based strategies:

DeepWalk (DW) [20]**:** uses random walks to generate network embeddings by exploring the local structure of graphs. Using short random walks, the algorithm captures neighborhood associations by generating sequences of nodes that mimic the connectivity patterns of the graph, similar to sentences in natural language processing. These sequences are subsequently input into a skip-gram model which projects them into a lower-dimensional vector space. The Skip-Gram model predicts neighboring nodes within a specified context window, constructing dense vector representations for each node.

Laplacian Eigenmaps (LEM) [21]**:** is a matrix factorization technique based on spectral analysis to generate embeddings. The method takes advantage of the geometric structure of the network to derive node embeddings. Initially, the adjacency matrix (A) and the degree matrix (D) are computed. These matrices are used to formulate the normalized Laplacian matrix: $N = D^{-1/2}AD^{-1/2}$. Subsequently, the algorithm identifies the k smallest non-zero eigenvalues and their corresponding eigenvectors. These truncated eigenvectors serve as the node embeddings, facilitating the preservation of local neighborhood structures.

Modularity-aware Graph Autoencoders (GAE) [22]**:** is a deep learning algorithm designed to reconstruct the adjacency matrix and other features related to the network through an auto-encoder framework for learning embedding. It incorporates a novel modularity-inspired regularizer that can be adjusted according to different reconstruction objectives, thereby capturing various aspects of network architecture. The model training process focuses on maximizing the modularity of the embeddings while minimizing the reconstruction error.

2.3 Embedding-Based Link Prediction

Link prediction through node embedding techniques is illustrated in Fig. 1. Initially, we extract the node embedding for each node in the network. Then, we derive edge embeddings by performing element-wise multiplication on the embeddings of node pairs. This process is executed for all possible node pairs to obtain their respective edge embeddings. Subsequently, a random sample (80%) of edge and non-edge embeddings are selected to form the training set. Following this, a Logistic Regression Classifier is trained using the generated edge embeddings. Ultimately, the trained model is capable of assessing any potential edge – regardless of its presence in the original graph – and computing its likelihood probability.

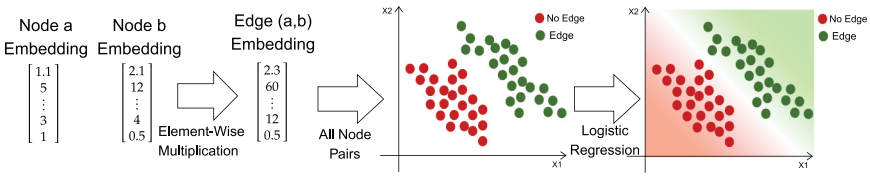


Fig. 1. An illustration of using node embedding for link prediction.

3 Backbone Extraction Using Embedding Link Prediction Methods

Inspired by statistical backbone extraction methods, which identify edges that significantly deviate from an established null model, our methodology focuses

on edges that diverge from the expectations set by the link prediction algorithm. Specifically, we extract a backbone by selecting the edges with the lowest probabilities according to the link prediction algorithm. Once the embedding is extracted and the model is trained, the backbone is derived as follows:

1. For each link in the network, compute its probability feeding the edge embedding into the trained classifier.
2. Sort the edges in ascending order based on their probabilities.
3. Select a specified fraction of the lowest-probability edges and remove the remaining edges along with any subsequent isolated nodes to construct the backbone.

4 Data and Methods

4.1 Dataset

The study uses four real-world networks from various domains (infrastructural, social, and character).

- **U.S. Air Transportation** [17]**:** Nodes represent US airports, edges correspond to direct flights, and edge weights indicate the number of passengers traveling by the corresponding route in the year 2006 symmetrized to produce an undirected representation.
- **World Air Transportation** [23]**:** Nodes represent world airports, edges correspond to direct flights, and edge weights indicate the number of flights operated by various airlines from May 17 to May 22, 2018.
- **American Elementary School** [24]**:** Nodes represent Kindergarten through sixth grade students at a suburban Utah elementary school, with edge weights indicating the total interactions between student pairs during January 31st and February 2, 2013.
- **Les Misérables** [25]**:** Nodes represent characters from Victor Hugo's "Les Misérables", with edges illustrating co-appearances in the same chapter. Edge weights denote the number of such co-appearances.

Table 1 reports their basic topological properties.

Table 1. The network topological features: $|N|$, $|E|$, $\langle k \rangle$, and ρ are the number of nodes, edges, average degree, and density.

Network	$\|N\|$	$\|E\|$	$\langle k \rangle$	ρ
US Air Transportation	380	9678	50.937	0.134
US-ES	339	16,546	97.61	0.289
World Air Transportation	2,729	16,654	12.205	0.004
Les Misérables	77	254	6.59	0.087

4.2 Backbone Evaluation Measures

Below, we specify the evaluation metrics used in the experiment.

- **Reachability:** This measure determines the level of connectivity among node pairs in a network. It is calculated as the proportion of node pairs that can interact: $\mathrm{R} = \frac{1}{n(n-1)} \sum_{i \neq j \in G} R_{ij}$, where n represents the number of nodes, and $R_{ij} = 1$ if there is a path between nodes i and j, otherwise $R_{ij} = 0$. Reachability scores range from 0 to 1, with $R = 1$ signifying complete connectivity and $R = 0$ representing total isolation.
- **Comparing Edge and Degree Distributions:** The Two-Sample Kolmogorov-Smirnov test compares the weight and degree distributions between the original network and the extracted backbones. This test assesses whether the distributions of two datasets are identical. The KS statistic, D, indicates the maximum difference between the Empirical Cumulative Distribution Functions (ECDFs) of the two datasets.

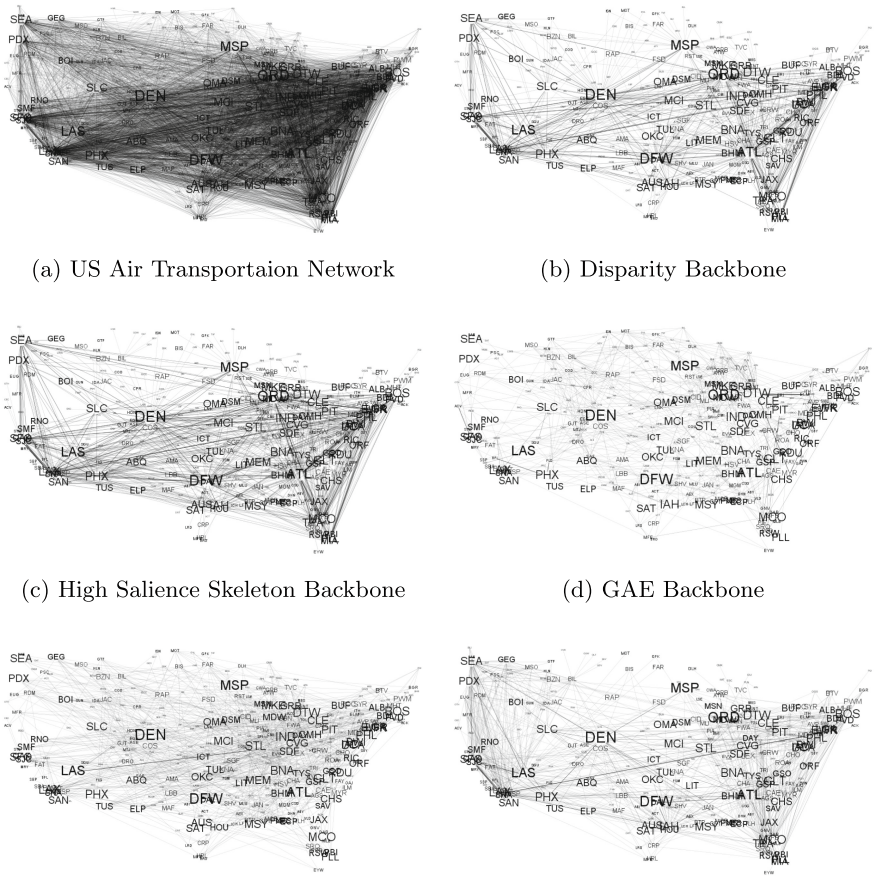
5 Experimental Results

Initially, we analyze the US air transportation backbone extracted with approximately 10% of its original links to qualitatively assess how well these approaches preserve essential structural features, like the hub-and-spoke topology of the US air transportation network. Next, we extract backbones for different edge fractions and measure properties such as the fraction of isolated nodes, reachability, degree, and weight distributions.

5.1 Investigating the US Air Transportation Backbones with a Fixed Backbone Size

The hub-and-spoke model is a dominant strategy in the airline industry, used by conventional airlines to optimize their operations [26]. This approach designates a central hub airport as the primary node through which flight routes are organized. By channeling passengers through these central hubs, airlines can efficiently connect multiple destinations, thus optimizing aircraft efficiency and enhancing the overall connectivity of the route network. Figure 2 depicts the structure of the US Air Transportation network and its backbones, which encompass approximately 10% of the total edges. In this visualization, the node sizes are proportional to their respective degrees, while the edge widths represent the edge weights.

The Disparity filter (Fig. 2b) method focuses on preserving edges that have significantly higher weights compared to others at the same node. This technique extracts a backbone with a single connected component. Notably, it eliminates approximately 15% of the airports, predominantly those categorized as spoke airports. This method clearly reveals the hub-and-spoke topology employed by airline companies. We can distinctly observe primary hubs in the United



(a) US Air Transportaion Network

(b) Disparity Backbone

(c) High Salience Skeleton Backbone

(d) GAE Backbone

(e) LEM Backbone

(f) Deepwalk Backbone

Fig. 2. The US Air Transportation network and the extracted backbones using different methdos. Note, the node sizes are proportional to their respective degrees, while the edge widths represent the edge weights.

States, such as DEN (Denver International Airport), ORD (O'Hare International Airport), DFW (Dallas/Fort Worth International Airport), ATL (Hartsfield-Jackson Atlanta International Airport), and LAX (Los Angeles International Airport), which are interconnected with smaller, regional spoke airports.

The High Salience Skeleton filter (Fig. 2c) emphasizes edges (connections) that frequently occur in the shortest paths connecting all nodes within the network. In the context of the US air transportation network, this filter preserves connections that are most frequently traversed by passengers traveling between various locations. Consequently, this leads to an emphasis on connections from local airports to regional hubs. Notably, there is a higher concentration of connections between the hubs used by the airline companies. For exam-

ple, there are observed connections from Denver to major East Coast hubs such as BOS (Boston), PHL (Philadelphia), CLT (Charlotte), ATL (Atlanta), MCO (Orlando), TPA (Tampa), FLL (Fort Lauderdale), and MIA (Miami).

Within the framework of link prediction techniques, we are conserving edges with the lowest probabilities, which are least predictable by the employed technique. Consequently, these edges connect the least similar nodes based on the embedding used.

The GAE backbone (Fig. 2d) retains approximately 90% of the nodes while removing about 10% of the airports, predominantly the peripheral or 'spoke' airports. In this backbone, connections between hub airports are absent. Instead, there are connections from spoke airports to the hubs and some regions exhibit high connectivity density. For example, to the west of Denver, airports such as HDN, EGE, ASE, GUC, MTU, GJT, TEX, and DRO form a densely connected cluster. Conversely, there exists another dense region in the north including airports like RST, LSE, ALO, DBQ, EAU, CWA, RHI, IMT, ESC, PLN, TVC, GRB, and ATW.

The LEM backbone (Fig. 2e) is similar to the GAE backbone, sharing 50% of the edges. However, unlike the GAE backbone, the LEM backbone does not exhibit dense regions. The edges primarily represent far non-regional connections. For instance, Denver (DEN) with Albany (ABY), Seattle (SEA) with San Jose (SJT) and Elmira (ELM), and Boston (BOS) with Roswell (ROW).

The DeepWalk backbone (Fig. 2f) preserves approximately 75% of the airports. Interestingly, we observe edges connecting different parts of the United States. For instance, Seattle (SEA) connects the North with the South through Las Vegas (LAS), Sacramento (SMF), Santa Ana (SNA), Burbank (BUR), and Palm Springs (PSP). Concurrently, it connects the West to the East through Atlanta (ATL), Charlotte (CLT), Houston (IAH), Philadelphia (PHL), and Pittsburgh (PIT). Additionally, Denver (DEN) in the middle connects to airports in every direction: Seattle in the Northwest, Boston in the Northeast, Minneapolis (MSP) in the North, Phoenix (PHX) and San Antonio (SAT) in the South, and Orlando (MCO) and Tampa (TPA) in the Southeast. Thus, the hub at DEN holds the U.S. air transportation network together, distinguishing it from other nodes in the network.

5.2 Comparing Backbone's Topological Properties with Varying Backbone Size

Isolated Nodes Fraction: The isolated node fraction quantifies the proportion of nodes excluded by the backbone. Specifically, it is the ratio of isolated nodes (those not connected to any other node) to the total number of nodes.

In all panels of figure Fig. 3, the high-salience skeleton (HSS) consistently results in the lowest isolated node fraction. However, the performance of other methods varies depending on the network. For the US-ES network, the disparity filter shows results comparable to those of HSS. While both the GAE and LEM catch up after an edge fraction of 0.3. In the global air transportation network, LEM outperforms other algorithms (excluding HSS) for edge fractions less than

0.5. However, Deepwalk becomes more effective for larger edge fractions. Finally, in the Les Misérables network, GAE is the most effective algorithm after HSS, followed by LEM.

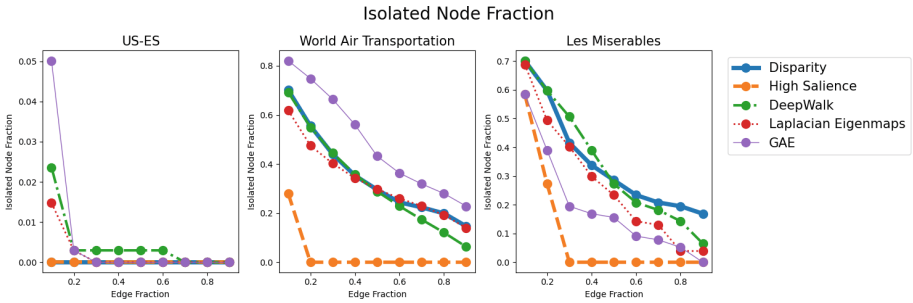


Fig. 3. Isolated node fraction as a function of the preserved edge fraction for each backbone extraction method.

Reachability: Reachability refers to the ability to traverse from one node to any other node within the network. A higher number of isolated nodes or disconnected components in the network leads to decreased reachability. Consequently, we anticipate an inverse relationship between the findings of the preceding section and node reachability. As illustrated in Fig Fig. 4, the overall ranking of the methods remains consistent except the US-ES network. In this case, the disparity filter exhibits a lower ranking and performs the poorest when a small fraction of edges is considered. This outcome implies that the disparity filter did not result in the isolation of numerous nodes but rather generated multiple small, disconnected components.

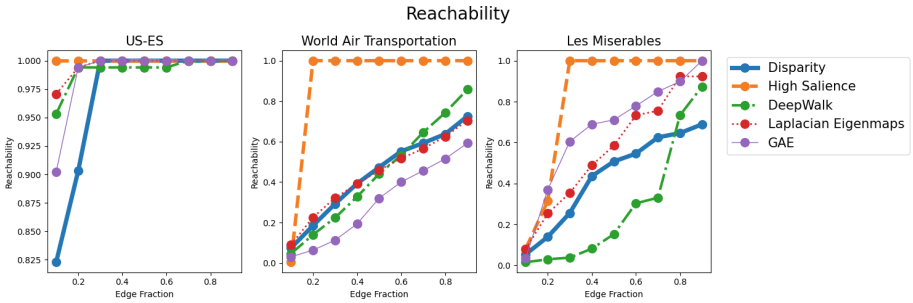


Fig. 4. Node reachability as a function of the preserved edge fraction for each backbone extraction method.

Weight Distribution: The weight scales denote the strength of connections and relationships between nodes. It is important for an optimal filtering technique to preserve these weights across different scales to maintain the integrity of the original network's weight distribution. The Kolmogorov-Smirnov (KS) test measures the distance between two distributions. A smaller KS distance implies that the two distributions are more similar. In Figure Fig. 5, we observe that in all panels, the DeepWalk method maintains the weight distribution of the original network more closely. Conversely, the Disparity method exhibits the greatest deviation from the original weight distribution across all networks. The behavior of the other methods varies and lacks consistency. In the first panel, concerning the US-ES network, DeepWalk is closely followed by GAE and LEM, which outperform both HSS and Disparity, especially for small edge fractions. In the second panel, HSS performs well for edge fractions below 0.3 but worsens beyond that point, ultimately performing the worst. The trends in the third panel are similar to those in the second panel. HSS starts strong, but worsens over time, though not as severely as in the global transport network case. GAE and then LEM follow HSS in performance.

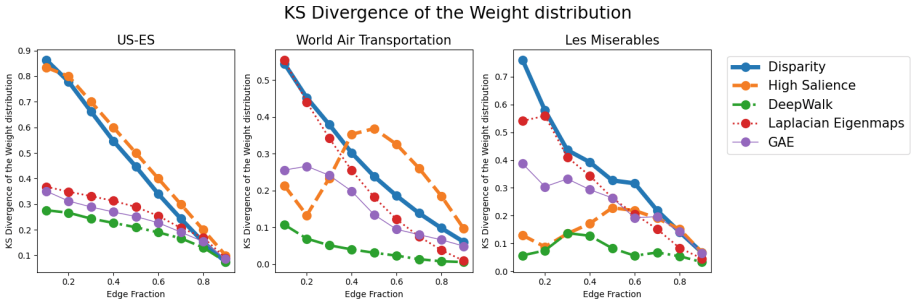


Fig. 5. The Kolmogorov-Smirnov (KS) static between the original and backbone weight distributions as a function of the preserved edge fraction for each backbone extraction methods

Degree Distribution: The degrees of nodes within a network serve as key indicators of the connectivity and influence of these nodes. A robust filtering method maintains various degrees, preserving the original degree distribution. Consistent with the previous experiment, the Kolmogorov-Smirnov (KS) test is employed to compute the distance between the two distributions. In Figure Fig. 6, all methods appear relatively similar. However, it is clear that the LEM, GAE, and DeepWalk methods outperform the HSS algorithm and the disparity filter within the US-ES network. For the air transport network, the HSS and LEM methods generate the best results up to an edge fraction of 0.5. Beyond this point, DeepWalk demonstrates superior performance. Notably, the disparity filter also shows competitive results. Conversely, the GAE method exhibits the poorest performance in this context. In the third panel, determining a clear

outperformer is more challenging. All algorithms, except for HSS, excel at certain points along the edge fraction continuum. GAE and LEM methods dominate for edge fractions below 0.3. At an edge fraction of 0.3, the disparity filter surpasses the other methods, and beyond this threshold, DeepWalk becomes the method that most effectively preserves the degree distribution.

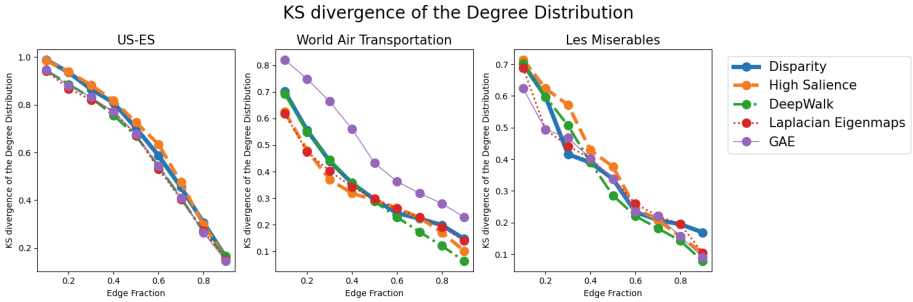


Fig. 6. The Kolmogorov-Smirnov (KS) static between the original and backbone degree distributions as a function of the preserved edge fraction for the backbone extraction method.

6 Conclusion

This study investigates embedding-based link prediction methods for backbone extraction in complex networks. Our research demonstrates that using the edge probability (existence of an edge) of embedding-based link prediction algorithms enables identification of the most significant edges.

A comparative analysis with traditional backbone extraction methods reveals different aspects of the hub-and-spoke business model used by airline companies in the US. Specifically, the Disparity and High Salience Skeleton backboens predominantly identify the hub-and-spoke topology, while the GAE and LEM methods uncover connections between local airports and regional hubs. The GAE method reveals clusters of dense connections within the same regions, whereas the LEM method identifies additional connections between non-regional airports. The Deepwalk backbone method reveals significant connections to Denver, the central hub of the United States, and links connecting cardinal and ordinal directions within the United States.

Examining the properties indicates that no single method excels in all aspects. The High Salience Skeleton backbone is the most effective in maintaining network nodes and reachability. In contrast, GAE, LEM, and Deepwalk methods compete effectively with the Disparity backbone in maintaining nodes and reachability. Additionally, the Deepwalk backbone excels in preserving weight distribution, while the LEM backbone is more effective at maintaining degree distribution. Future research should broaden the exploration of alternative link prediction approaches for backbone extraction beyond those based on node embedding.

Acknowledgment. This material is based upon work supported by the Agence Nationale de Recherche under grant ANR-20-CE23-0002.

References

1. Zignani, M., Quadri, C., Gaito, S., Cherifi, H., Rossi, G.P.: The footprints of a mastodon: how a decentralized architecture influences online social relationships. In: IEEE INFOCOM 2019, pp. 472–477. IEEE (2019)
2. Erum, I., Malick, R.A.S., Ahmed, G., Cherifi, H.: Complex network analysis of mass violation, specifically mass killing. Entropy **24**(8), 1017 (2022)
3. Güell, O., Sagués, F., Serrano, M.Á.: Detecting the significant flux backbone of Escherichia coli metabolism. FEBS Lett. **591**(10), 1437–1451 (2017)
4. Kumar, P., Verma, P., Singh, A., Cherifi, H.: Choosing optimal seed nodes in competitive contagion. Front. Big Data **2**, 16 (2019)
5. Chakraborty, D., Singh, A., Cherifi, H.: Immunization strategies based on the overlapping nodes in networks with community structure. In: Nguyen, H.T.T., Snasel, V. (eds.) CSoNet 2016. LNCS, vol. 9795, pp. 62–73. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42345-6_6
6. Sow, A., Diallo, C., Cherifi, H.: Effects of vertical transmission and human contact on zika dynamics. Complexity **2022**(1), 5366395 (2022)
7. Sow, A., Diallo, C., Cherifi, H.: Interplay between vaccines and treatment for dengue control: an epidemic model. PLoS ONE **19**(1), e0295025 (2024)
8. Tilly, S., Livan, G.: Macroeconomic forecasting with statistically validated knowledge graphs. Expert Syst. Appl. **186**, 115765 (2021)
9. Diop, I.M., Cherifi, C., Diallo, C., Cherifi, H.: Revealing the component structure of the world air transportation network. Appl. Netw. Sci. **6**, 1–50 (2021)
10. Yassin, A., Haidar, A., Cherifi, H., Seba, H., Togni, O.: An evaluation tool for backbone extraction techniques in weighted complex networks. Sci. Rep. **13**(1), 17000 (2023)
11. Neal, Z.P.: backbone: an R package to extract network backbones. PLoS ONE **17**(5), e0269137 (2022)
12. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: Extracting backbones in weighted modular complex networks. Sci. Rep. **11**, 12 (2021)
13. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: Extracting modular-based backbones in weighted networks. Inf. Sci. **576**, 454–474 (2021)
14. Hmaida, S., Cherifi, H., El Hassouni, M.: A multilevel backbone extraction framework. Appl. Netw. Sci. **9**(1), 41 (2024)
15. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: A community-aware backbone extractor for weighted networks. In: 9th International Conference on Complex Networks and Their Applications, pp. p–3 (2020)
16. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: Link prediction techniques, applications, and performance: a survey. Phys. A Stat. Mech. Appl. **553**, 124289 (2020)
17. Serrano, M.A., Boguna, M., Vespignani, A.: Extracting the multiscale backbone of complex weighted networks. PNAS **106**, 6483–6488 (2009)
18. Grady, D., Thiemann, C., Brockmann, D.: Robust classification of salient links in complex networks. Nat. Commun. **3**, 864 (2012)
19. Cui, P., Wang, X., Pei, J., Zhu, W.: A survey on network embedding. IEEE Trans. Knowl. Data Eng. **31**(5), 833–852 (2018)

20. Qiu, J, Dong, Y., Ma, H., Li, J., Wang, K., Tang, J.: Network embedding as matrix factorization: unifying DeepWalk, LINE, PTE, and node2vec. In: International Conference on Web Search and Data Mining, pp. 459–467 (2018)
21. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. Adv. Neural Inf. Process. Syst. **14** (2001)
22. Salha-Galvan, G., Lutzeyer, J.F., Dasoulas, G., Hennequin, R., Vazirgiannis, M.: Modularity-aware graph autoencoders for joint community detection and link prediction. Neural Netw. **153**, 474–495 (2022)
23. Alves, L., Aleta, A., Rodrigues, F., Moreno, Y., Amaral, L.: Centrality anomalies in complex networks as a result of model over-simplification. New J. Phys. **22**, 01 (2020)
24. Yassin, A., Cherifi, H., Seba, H., Togni, O.: A modular network exploration of backbone extraction techniques. In: International Conference on Complex Networks and Their Applications, pp. 296–308 (2023)
25. Knuth, D.: The Stanford GraphBase. A Platform for Combinatorial Computing, vol. 1 (1993)
26. Yassin, A., Cherifi, H., Seba, H., Togni, O.: Exploring statistical backbone filtering techniques in the air transportation network. In: 2022 IEEE Workshop on Complexity in Engineering (COMPENG), pp. 1–8 (2022)

Deep Learning Based Link Prediction for Minimizing Maximum Betweenness

Aashish[1], Parul Diwakar[1], Venkata Narendra Kotyada[2], Rishi Ranjan Singh[1](✉), and Soumajit Pramanik[1]

[1] Department of Computer Science and Engineering, Indian Institute of Technology Bhilai, Durg, Chhattisgarh, India
{aashish,paruldiwa,rishi,soumajit}@iitbhilai.ac.in

[2] National Institute of Technology, Tadepalligudem, Andhra Pradesh, India
421202@student.nitandhra.ac.in

Abstract. Centrality measure is a tool to identify key nodes/edges in networks having higher importance/control/load etc. Betweenness centrality ranks nodes to locate the nodes with maximum control of information flow or nodes heavily loaded due to information flow happening through the shortest paths. The amount of traffic passing through a node is estimated by computing its betweenness centrality. The highest value of betweenness centrality observed among all the nodes in a network is termed as *maximum betweenness* of the network. We consider the problem of finding a missing link in a network, adding which minimizes the network's maximum betweenness. We propose a Graph Neural Network (GNN) based approach to predict such a missing link. Experiments are conducted on various types of synthetic networks to evaluate the performance of the proposed model.

Keywords: Graph Neural Network · Betweenness Centrality · Link Prediction

1 Introduction

Centrality measures are widely used in various domains, such as social networks, transportation networks, and biological networks, to identify key players, critical nodes, or important genes [19,21]. The positive interpretation of centrality scores signifies attributes like dominance, power, control, and influence. On the other hand, the negative interpretation reflects aspects such as vulnerability, susceptibility to failure, stress on nodes, and exposure to potential threats or infections. Identifying key players could help get an insight into the structure of an organization to find the most influential ones that hold more power in decision-making than others. Similarly, insights gained from analyzing key nodes can help us make informed strategic decisions, like prioritizing collaborations and partnerships for collective gain. Readers are referred to the books [5] on centrality measures for further details. Degree centrality, betweenness centrality, eigenvector centrality, and closeness centrality are the most commonly used centrality measures.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 237–249, 2025.
https://doi.org/10.1007/978-3-031-82439-5_20

In this paper, we focus on the betweenness centrality. We also refer to betweenness centrality as BC at times in this paper. Betweenness centrality ranks nodes to locate the nodes with maximum control of information flow or nodes heavily loaded due to information flow happening through the shortest paths. The amount of traffic passing through a node is estimated by computing its betweenness centrality. In other words, it helps to determine how frequently a node in the network is on the shortest path between all other pairs of nodes. The highest value of betweenness centrality observed among all the nodes in a network is termed as *maximum betweenness* of the network. It signifies the maximum load or traffic in a network, which can be a notion of the required amount of maintenance and resources for the proper functioning of the system represented by that network. Nodes with maximum betweenness scores also pose risks like network bottlenecks and congestions, unequal distribution of power imbalances, and vulnerability to attack and fragmentation. One of the possible ways to reduce the load on nodes is by diverting traffic through new pathways created by adding edges. We consider the problem of finding a missing link in a network, adding which minimizes the network's maximum betweenness. Identifying such an edge deterministically in a network turns out to be computationally expensive due to several rounds of betweenness score computation. Therefore, we leverage the deep learning models to solve our problem. We propose a Graph Neural Network(GNN) based approach to predict such a missing link with rank loss that attempts to predict a missing edge in a network, adding which can maximally reduce the maximum betweenness of the network. A solution to this problem could aid in developing better transportation systems and assist in making congestion management decisions.

1.1 Problem Definition

Let $G = (V, E)$ be a network, where V denotes the set of nodes, and E denotes the set of links. We use the terms link and edge, network and graph interchangeably throughout the paper. The betweenness centrality of a node v in network G is defined as $BC_G(v) = \sum_{s,t,v \in V \mid s \neq t, t \neq v, v \neq s} \frac{\sigma_{st}(v)}{\sigma_{st}}$ where $\sigma_{st}(v)$ denotes the number of shortest paths between node s and node t passing through node v. σ_{st} denotes the total number of shortest paths between node s and node t in network G. Betweenness Centrality can be computed using Brandes' algorithm [4], which is asymptotically the state-of-the-art algorithm for computing the exact betweenness scores of all nodes in a network. The time complexity of this algorithm is $\mathcal{O}(|V||E|)$ on unweighted networks and $\mathcal{O}(|V||E|+|V|^2 \log |V|)$ on weighted networks respectively, where $|V|$ and $|E|$ denote the numbers of nodes and links.

Given a network $G = (V, E)$, we construct a new network $G'_e = (V, E \bigcup \{e\})$ for some $e \in \overline{E}$, where $\overline{E}$ denotes the set of missing links in G. In the problem of minimizing the maximum betweenness of network G, we aim to identify a missing link from the set $\overline{E}$, adding which achieves the minimum maximum centrality

score of a network G. In other words, we aim to find that link e_{min} such that,

$$e_{min} = \arg\min_{e \in \overline{E}} [\max_{v \in V} [BC_{G'_e}(v)]].$$

The brute force method to identify such a link requires constructing $K = \binom{|V|}{2} - |E|$ many networks by adding one missing at a time and computing the betweenness centrality scores of all the nodes to find the maximum betweenness of all K networks. If Brandes' algorithm [4] is used to compute each round of betweenness centrality scores, the total time taken by the brute force algorithm is $\mathcal{O}(K \cdot (|V||E|)$ for unweighted networks and $\mathcal{O}(K \cdot (|V||E|+|V|^2 \log |V|))$ for weighted networks. Most of the real-world networks are sparse and have constant average degrees. In sparse networks, $|E| = \mathcal{O}(|V|) \implies K = \mathcal{O}(|V|^2)$. Therefore, in the sparse network, the brute force method takes $\mathcal{O}((|V|^3|E|)$ and $\mathcal{O}(K \cdot (|V|^3|E|+|V|^4 \log |V|))$ for unweighted and weighted networks respectively. This time complexity is high even for moderate-size real-world networks.

2 Related Works

Existing literature explores link prediction or centrality estimation problems as a stand-alone task. Though some work related to the maximization of centrality scores of a node exists, a gap persists regarding the strategic link prediction aimed at reducing centrality scores in a given network when the scores represent a negative notion. First, we review the literature on problems that focus on maximizing changes in centrality scores or other structural properties of nodes through the addition of missing links. Next, we address research on machine learning methods for centrality estimation. Finally, we discuss studies related to link prediction using deep learning and machine learning techniques.

Bergamini et al. [3]studied the link prediction problem aimed at maximizing a node's betweenness. Brokkelkamp et al. [6] studied flow maximization problem in a directed network when the weights of a few edges are allowed to be changed D'Angelo et al.[8] proposed two greedy approaches focused on the centrality maximization problem for coverage centrality.

An in-depth explanation of the principles and foundational mathematics behind the GNN model has been give by Scarselli et al. [20]. Wu et al. [24] carried out a comprehensive survey of various categories of GNNs, some benchmark datasets, and reviews on some state-of-the-art techniques, classifying the models into recurrent GNNs (RecGNNs), convolutional GNNs (ConvGNNs), graph autoencoders (GAEs), and spatial-temporal GNNs (STGNNs). Several studies have been conducted to approximate the betweenness score of nodes in networks using graph Learning techniques. A GNN base ranking model given in [10] predicts nodes with high betweenness centrality. An encoding stage maps each node's structural information into an embedding space which are then used in the decoding stage to compute the ranking scores. Mendonça et al. [16] used the degree of nodes as the only input feature to approximate node centrality.

Grando et al. [11] demonstrated that a regression model based on artificial neural networks performs well in estimating centrality scores. In [15], the authors propose a GNN model to approximate betweenness and closeness centrality in networks. This approach outperforms existing techniques in accuracy and efficiency, and its inductive nature allows it to generalize across varying graph structures. Recently, [12] proposed a GNN model to estimate critical links in a road network by ranking edges based on edge betweenness scores.

Graph learning techniques have proven to be a prominent tool in link prediction tasks. Dong et al. [7] introduced two algorithms (based on local information, based on global network data) for link prediction. A new recommendation framework was given by Papadimitriou et al. [17] for new friend connections in a social network. Ayoub et al. [1] proposed an innovative metric for link prediction that uses local similarity indices and betweenness centrality.

3 Methodology

In this section, we present our proposed framework for estimating the impact of missing edges in reducing the maximum betweenness centrality of a graph. As depicted in Fig. 1, our proposed framework consists of two stages -

- In stage 1, we leverage Graph Neural Networks (GNN) to learn node embeddings capturing the structural representation of the graph with the help of message passing and neighborhood aggregation techniques;
- In stage 2, we utilize two feed-forward Multi-Layer Perceptrons (MLP) to estimate the ranking of missing edges according to their contribution in reducing the maximum betweenness centrality of the graph.

In the following, we describe the details of the proposed stages one by one.

3.1 Stage 1

In this stage, with the help of a GNN, we extract features from the graph structure and produce node representations, also known as embeddings. In this work,

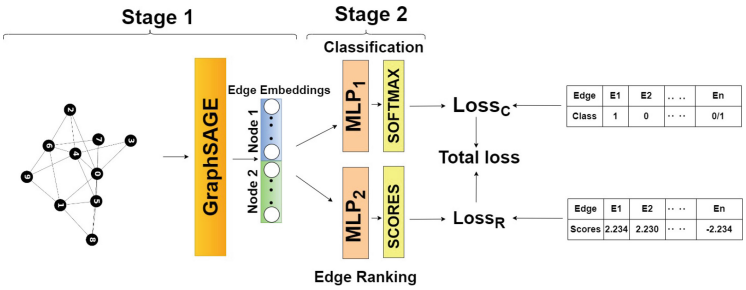


Fig. 1. Workflow of the proposed model.

we use the GraphSAGE [23] framework which relies upon inductive learning for generalizing to unseen nodes and graphs.

We denote the representation of each node v as h_v and initialize them randomly at the beginning. In the following, we describe how we update these embedding iteratively.

Aggregation: The aggregation step involves combining the states of all neighboring node v with node u in a specific manner.

$$h_{N(v)}^{(k)} = AGG_k(\{h_u^{(k-1)} | u \in N(v)\}) \tag{1}$$

where $h_u^{(k-1)}$ represents the feature vector (embedding) of neighbor of node u from the $(k-1)^{th}$ layer, $N(v)$ is the set of neighbors of node v and $h_{N(v)}^{(k)}$ is the aggregated feature vector for the neighborhood of node v at layer k. Several kinds of aggregation functions are supported by the GraphSAGE framework such as Mean aggregator, LSTM aggregator, and Pooling aggregator. In our work, we have used the Mean aggregator.

Updation: After aggregating the neighborhood features, GraphSAGE combines the node's own features with the aggregated neighborhood features to update their representations at the next layer.

$$h_v^{(k)} = \sigma(W^{(k)} \cdot CONCAT(h_v^{(k-1)}, h_{N(v)}^{(k)})) \tag{2}$$

where $h_v^{(k-1)}$ is the feature vector of node v from the $k-1^{th}$ layer and $W^{(k)}$ is a learnable weight matrix. σ is a non-linear activation function. The idea behind concatenation is to preserve both, the node's features and the aggregated neighborhood features when updating the node's representation.

In order to obtain the representation for each missing edge (u, v), we concatenate node embeddings obtained from the final layer of the GNN i.e. $E_{u,v} = CONCAT[h_u^{(n)}, h_v^{(n)}]$ and pass it as input to the feed-forward Multi-Layer Perceptron (MLP) networks in stage 2.

3.2 Stage 2

In this stage, we utilize two MLP networks - MLP_1 for classifying the edge (binary classification) based on whether it has the potential to lower the maximum betweenness centrality of the graph or not (1/0); and MLP_2 for predicting the score for each edge, based on which it can be ranked.

MLP_1**:** Here, we consider $E_{u,v}$ as the input, pass it through a series of hidden layers and produce the probabilities of it falling into the positive and negative classes utilizing a Softmax[14] function at the end.

MLP_2**:** We require a score for each missing edge to rank the missing edges. With this goal, the edge embeddings ($E_{u,v}$) are fed into MLP_2 to generate scores for all missing edges.

These predicted labels and scores are used further to calculate the total loss for each missing edge. While predicting the ranking of edges, we combine the two aforementioned MLPs in the following way. For every missing edge, we classify it using MLP_1 and rank it using MLP_2 if it is labeled as a positive edge by MLP_1.

3.3 Loss Calculation

Classification Loss: In MLP_1, a binary cross entropy loss is used for classification that compares each of the predicted probabilities to the actual class output (labels), and can be either 0 or 1.

$$Loss_C = -[y \times \log(\hat{y}) + (1 - y) \times \log(1 - \hat{y})] \tag{3}$$

where y denotes ground truth labels(either 0 or 1) and $\hat{y}$ denotes the predicted probability of the positive class (ranging between 0 to 1).

Ranking Loss: A ranking loss function is used in MLP_2 to estimate the loss due to the difference in the ranking predicted by the proposed model compared to the ground truth ranking. In such a scenario, typically pairwise ranking loss functions have been used for recommendation systems to rank or rate products or users [15]. However, in our case, we propose a custom 'triplet' margin loss, which compares the ranking for three edges at a time, serving as an extension of the pairwise margin loss. The proposed triplet margin ranking loss function for a triplet of three edges {m,n,o} is defined as follows:

$$\begin{aligned} Loss_R = \max \Big[0, -I_{m,n} \times (s_m - s_n) + \mathrm{M}, -I_{n,o} \times (s_n - s_o) + \mathrm{M}, \\ -I_{o,m} \times (s_o - s_m) + \mathrm{M}\Big] \end{aligned} \tag{4}$$

where, s_m, s_n, and s_o are predicted ranking scores for edges m, n, and o respectively. $\hat{s_m}$, $\hat{s_n}$, and $\hat{s_o}$ are the ground truth scores (the amount by which it reduces maximum betweenness centrality). $I_{m,n}$ is considered as 1 if $\hat{s_m} > \hat{s_n}$, and -1, otherwise. The Margin M is set to 1 to allow some flexibility. Finally, we combine both loss functions using unweighted summation that incorporates both classification and ranking objectives, i.e., Total Loss = $Loss_C + Loss_R$. We backpropagate the total loss via both MLPs and the GNN.

4 Experimental Setup

In this section, we discuss the way we generate our datasets along with various training parameters and evaluation metrics used for our experiments.

4.1 Data Generation

We opt for two sets of synthetic networks generated from the Barabási-Albert (BA) [2] model which gives scale-free networks with preferential attachment phenomenon and the Erdős Rényi (ER) [9] model which gives structureless, purely

Table 1. Dataset

BA		ER		Tree		Path	
Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
# nodes	{165, 180, 190, 200}	# nodes	{165, 180, 190, 200}	# nodes	{180, 190, 200, 210, 220}	# nodes	[160, 250]
m-value	{3, 4, 5, 6 }	Probability	[0.03, 0.0648]	k factor	{3, 4, 5, 6, 7, 8}		
seed values	[0, 1000]	seed values	[0, 1000]				
# Graphs	100	# Graphs	100	# Graphs	30	# Graphs	10

random networks with different diameters, to encompass a diverse range of network topologies. 20 sets of graphs (consisting of 10 graphs each) are generated for each kind of topology. Apart from these we also use a few sets of Tree and Path networks (details in Table. 1). For each network, we systematically introduce missing edges into the network, one at a time, and calculate the difference between the maximum betweenness centrality of the modified network and the original network (see Fig. 2).

4.2 Implementation Details

We implement our model using the PyTorch framework on a system with NVIDIA Tesla T4 GPU and Intel Xeon CPU. The edge embedding size is set to 64 whereas the hidden state dimension is kept as 32. We train our model for maximum 225 epochs (with early stopping) using an Adam optimizer with a learning rate of 0.0085 and maintaining a dropout rate of 0.5 (to prevent overfitting). The number of GNN layers is set to one less than the diameter for BA, and ER graphs, and for paths and trees, it is set to 3, and the number of hidden layers for MLPs is set to 3. All the parameters mentioned above are chosen empirically.

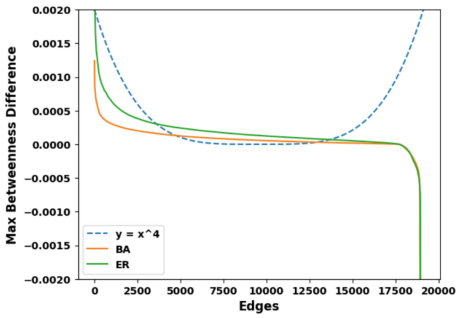


Fig. 2. Edges sorted as per their contribution towards reducing maximum betweenness centrality for BA and ER networks and sampling distribution.

The dataset is split into training, validation, and test sets following a 64:16:20 ratio. The edge sample size for training is set to be 5000, and the triplet loss is calculated by sampling triplets judicially, with sample sizes 10, 15, and 20 times the edge sample size.

As observed from Fig. 2, for most of the edges the maximum betweenness centrality difference values are found to be 0 or very close to 0. These edges do not help in minimizing max BC. Therefore, while training, we use a sampling distribution ($y = x^4$) that samples edges from the left-most and right-most regions of the plot with high probability. For ranking, we use random sampling to sample triplets from predicted ranks during triple loss computation.

4.3 Metrics

We evaluate our model based on the standard correlation coefficient (Pearson's Correlation Coefficient (PC) [18]) and ranked correlation coefficients (Kendall's Coefficient (KC) [13], and Spearman's Coefficient (SC) [22]). Apart from them, we use Mean Average Precision@k (mAP@k) as defined below.

Mean Average Precision@k (mAP@k) It is a metric used to evaluate the quality of ranked lists of edges in graphs.

$$\text{mAP@k} = \frac{1}{N_{\text{graphs}}} \sum_{i=1}^{N_{\text{graphs}}} \text{AP}_i@k$$

where N_{graphs} is the total number of graphs considered in the evaluation. $\text{AP}_i@k$ is the average of precision values for top k model predicted edges for the ith graph. It is calculated as $\text{AP@k} = \frac{1}{k}\sum_{j=1}^{k} \text{P@j} \times \text{Rel}(j)$ where $P@j$ denotes the precision at j^{th} position i.e. the fraction of relevant edges among the top j model predicted edges and $\text{Rel}(j)$ is an indicator function that returns 1 if the j-th edge is relevant (among the top k ground truth edges) and 0 otherwise. mAP emphasizes the importance of capturing as many relevant edges as possible in the top k, without significantly penalizing the model for slight inconsistency in the ranking order.

5 Results and Discussion

5.1 Performance Metrics

BA synthetic networks follow scale-free degree distribution (a few highly connected nodes/hubs dominate) and closely mirror real-world networks such as social and transport networks. Figure 3a depicts that the model has achieved an impressive score of mAP 0.82 in the top 10 edges and 0.92 in the top 100 edges for BA networks, which indicates how accurately the top k ranked edges are predicted. Path and tree are networks with higher diameters in which the addition of an edge between distant nodes causes a major reduction in maximum betweenness. Therefore, the model performed significantly better and has mAP

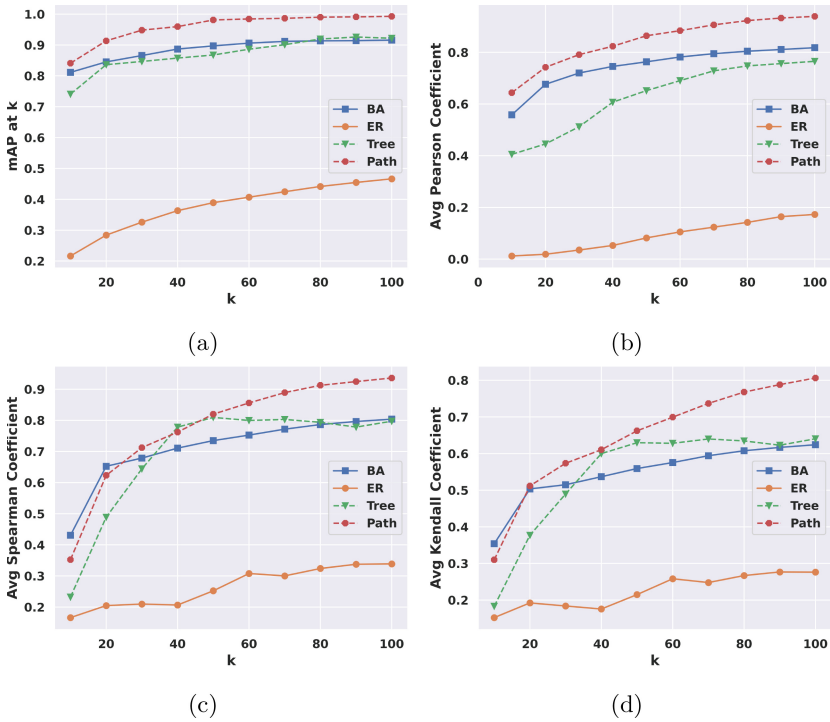


Fig. 3. (a) Average mAP at k for all graphs. (b) Average Pearson's Coefficient coefficient (PC) of all graphs. (c) Average Spearman's Coefficient (SC) of all graphs. (d) Average Kendall's Coefficient (KC) of all graphs.

values of 0.84 in the top 10 and 0.99 in the top 100 over path networks. On tree networks, the model performs comparably to BA networks with mAP@10 as 0.74 and mAP@100 as 0.92. The model does not perform well on ER networks. The performance of our model in predicting the best missing edge in terms of mAP@1 is found to be 0.60, 0.81, and 0.35 for BA, path, and tree networks, respectively. Even if the best edge is not the top-predicted edge, it is mostly present in the top 10 predicted edges by the model on BA and path networks.

Figure 3b 3c 3d show the performance of our model based on correlation-based evaluation metrics for all graphs. PC checks the linear relationship between predicted edge scores and the corresponding actual reductions in maximum betweenness. Figure 3b shows that the proposed model achieves a high average PC score of 0.82 for the top 100 edges of BA networks. The PC scores for tree and path networks are 0.78 and 0.92, respectively. In contrast, the correlation value is notably low 0.18 for ER networks. We use KC and SC to capture the performance of the proposed model in maintaining the rank order of predicted edges to their actual rank in the ground truth. Figure 3c shows that the model achieves high SC of 0.81, 0.81, and 0.93 for the top 100 edges in BA, tree, and

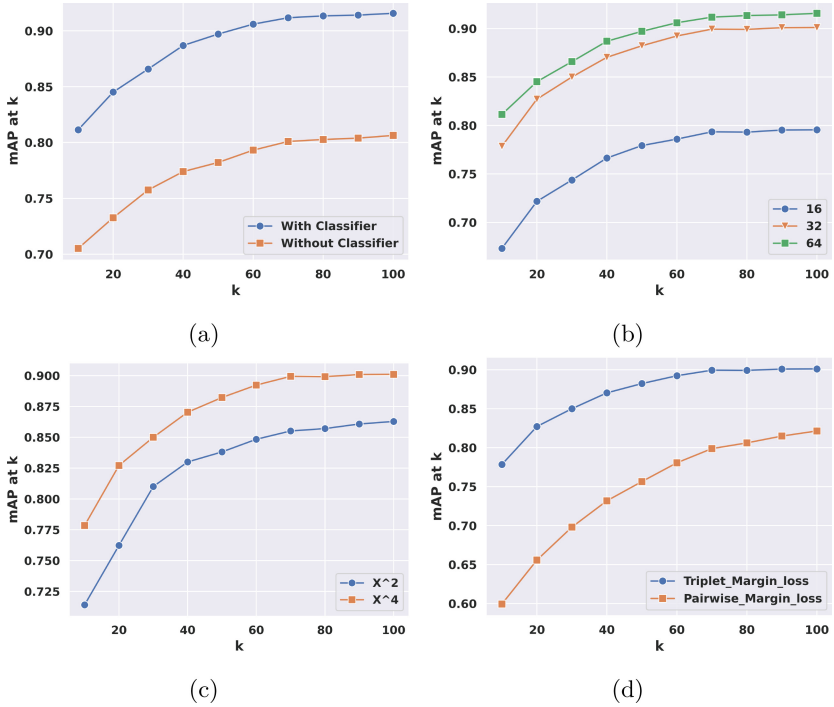


Fig. 4. (a) Average mAP@k for BA graphs with and without MLP_1. (b) Mean Average Precision versus top k edges for edge embedding dimension. (c) Mean Average Precision versus top k edges for sampling x^2 and x^4 distribution. (d) Mean Average Precision versus top k edges for loss functions.

path networks whereas Fig. 3d shows that the model achieves high KC of 0.63, 0.65, and 0.81 for the top 100 edges in BA, tree, and path networks.

PC, KC, and SC show strong agreement with the mAP metric. The achieved performance in BA is comparable with large-diameter networks such as paths and trees. This ascertains the applicability of our model for large-scale real-world networks. The lack of inherent hierarchy, hubs, large-size diameter, and community structures in ER networks makes it difficult for the model to identify critical edges that can significantly reduce the maximum betweenness of networks.

5.2 Ablation Test and Hyperparameter Tuning

We perform an ablation test to emphasize the need of using both MLPs in our proposed model. Although, intuitively it seems sufficient to generate the ranking of edges using only MLP_2, Fig. 4a highlights that it leads to a significant performance drop in the case when we do not use the classifier MLP (MLP_1) and the corresponding classification loss.

The proposed model is sensitive to various tunable parameters such as edge embedding size. Experiments are conducted while varying the edge embedding dimension as [16, 32, 64]. Figure 4b shows that our model performs the best with 64-dimensional edge embedding. Experiments with the following sampling distributions are conducted to compare how different sampling strategies of edges affect the mAP: x^2 and x^4. The comparison is shown in Fig. 4c. The distribution x^4 tends to sample more extreme edges, leading to better results as the model can differentiate and rank positive edges effectively. To analyze the effect of the ranking loss functions, we compare the proposed triplet margin loss with the pairwise ranking loss [15]. Figure 4d shows that triplet margin loss outperforms pairwise loss in terms of mAP.

6 Conclusion

We considered the problem of identifying missing links that may lead to a maximum reduction in the maximum betweenness of a network. We proposed a GNN based approach that learns to capture the complex, underlying structure of networks and predicts such missing links. The proposed model uses two MLPs - one for classifying the important edges and the other one for ranking them. Both of these are optimized based on a joint loss function combining both objectives. The proposed model demonstrates high precision in predicting the top k edges in BA networks, which resemble real-world networks in terms of degree distribution. Interestingly, the achieved precision in BA is found to be comparable with large diameter networks such as paths and trees where finding such missing links is relatively easier. On the contrary, the model does not perform well for the ER networks due to a lack of inherent hierarchy, hubs, large diameter, and community structures in ER networks. One of the future directions of this work would be to evaluate the proposed model for more synthetic and real-world networks of large size. Further, it would also be interesting to adapt the existing machine learning models for estimating node/edge betweenness scores for the considered problem and compare them with the proposed model.

Acknowledgements. This work was supported in part by a SERB Core Research Grant(CRG/2023/007610).

References

1. Ayoub, J., Lotfi, D., Hammouch, A.: Link prediction using betweenness centrality and graph neural networks. Soc. Netw. Anal. Min. **13**(1), 5 (2022)
2. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
3. Bergamini, E., Crescenzi, P., D'angelo, G., Meyerhenke, H., Severini, L., Velaj, Y.: Improving the betweenness centrality of a node by adding links. J. Exp. Algorithmics **23**, 1–32 (2018)

4. Brandes, U.: A faster algorithm for betweenness centrality. J. Math. Sociol. **25**(2), 163–177 (2001)
5. Brandes, U., Erlebach, T. (eds.): Network Analysis. LNCS, vol. 3418. Springer, Heidelberg (2005). https://doi.org/10.1007/b106453
6. Brokkelkamp, R., Polak, S., Schäfer, G., Velaj, Y.: Approximate pricing in networks: how to boost the betweenness and revenue of a node. In: 30th International Symposium on Algorithms and Computation (ISAAC 2019). Schloss-Dagstuhl-Leibniz Zentrum für Informatik (2019)
7. Dong, L., Li, Y., Yin, H., Le, H., Rui, M.: The algorithm of link prediction on social network. Math. Probl. Eng. **2013** (2013)
8. D'Angelo, G., Olsen, M., Severini, L.: Coverage centrality maximization in undirected networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 501–508 (2019)
9. Erdős, P., Rényi, A.: On the strength of connectedness of a random graph. Acta Math. Acad. Scientiarum Hungarica, 261–267 (1961). https://doi.org/10.1007/BF02066689
10. Fan, C., Zeng, L., Ding, Y., Chen, M., Sun, Y., Liu, Z.: Learning to identify high betweenness centrality nodes from scratch: a novel graph neural network approach. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 559–568 (2019)
11. Grando, F., Granville, L.Z., Lamb, L.C.: Machine learning in network centrality measures: tutorial and outlook. ACM Comput. Surv. (CSUR) **51**(5), 1–32 (2018)
12. Jana, D., Malama, S., Narasimhan, S., Taciroglu, E.: Edge-based graph neural network for ranking critical road segments in a network. PLoS ONE **18**(12), e0296045 (2023)
13. Kendall, M.G.: A new measure of rank correlation. Biometrika **30**(1–2), 81–93 (1938)
14. Liao, B., Jungang, X., Lv, J., Zhou, S.: An image retrieval method for binary images based on DBN and Softmax classifier. IETE Tech. Rev. **32**(4), 294–303 (2015)
15. Maurya, S.K., Liu, X., Murata, T.: Graph neural networks for fast node ranking approximation. ACM Trans. Knowl. Discov. Data (TKDD) **15**(5), 1–32 (2021)
16. Mendonça, M.R.F., Barreto, A.M.S., Ziviani, A.: Approximating network centrality measures using node embedding and machine learning. IEEE Trans. Netw. Sci. Eng. **8**(1), 220–230 (2020)
17. Papadimitriou, A., Symeonidis, P., Manolopoulos, Y.: Fast and accurate link prediction in social networking systems. J. Syst. Softw. **85**(9), 2119–2132 (2012)
18. Pearson, K.: Notes on the history of correlation. Biometrika **13**(1), 25–45 (1920)
19. Saxena, A., Iyengar, S.: Centrality measures in complex networks: a survey. arXiv preprint arXiv:2011.07190 (2020)
20. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE Trans. Neural Netw. **20**(1), 61–80 (2008)
21. Singh, R.R.: Centrality measures: a tool to identify key actors in social networks. In: Principles of Social Networking: The New Horizon and Emerging Challenges, pp. 1–27 (2022)

22. Spearman, C.: The Proof and Measurement of Association Between Two Things (1961)
23. Leskovec, J., Hamilton, W.L., Ying, R.: Inductive representation learning on large graphs. In: 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA (2017)
24. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learn. Syst. **32**(1), 4–24 (2020)

Global Link Prediction Strategies for Backbone Extraction in Real-World Networks

Ali Yassin[1(✉)], Hocine Cherifi[2], Hamida Seba[3], and Olivier Togni[1]

[1] Laboratoire d'Informatique de Bourgogne, Université de Bourgogne, Dijon, France
aliyassin4@hotmail.com
[2] ICB UMR 6303 CNRS, Université de Bourgogne, Dijon, France
[3] Univ Lyon, UCBL, CNRS, INSA Lyon, LIRIS, UMR5205, 69622 Villeurbanne, France

Abstract. Network backbone extraction is essential for simplifying complex systems while preserving key structural characteristics. Traditional methods often compromise important properties like edge weight distribution. This study introduces an innovative approach by applying global similarity-based link prediction techniques to extract network backbones. Through a comparative analysis using the world air transportation network, we reveal how different methods reflect various aspects of the hub-and-spoke model employed by airlines. Specifically, the Katz backbone method effectively combines local and global connections, while the High Salience Skeleton method excels in maintaining network nodes and reachability. Based on three real-world networks, our results show that the Katz backbone preserves weight distribution most effectively, whereas the SP and Disparity backbones maintain degree distribution more accurately. These findings suggest that the backbone extraction technique should be tailored to the specific network property of interest.

Keywords: Complex Networks · Backbone Extraction · Link Prediction · Filtering Techniques · Network Compression · Graph Summarization · Sparsification · Edge Similarity

1 Introduction

Extracting the backbone of a network—a simplified representation that retains the most critical connections—is crucial for effective network analysis. This process is fundamental in various domains, such as social network analysis [1,2], bioinformatics [3,4], epidemiology [5–7] and transportation systems [8,9]. Indeed, understanding the core structure of a network can lead to valuable insights and more efficient operations. This trade-off presents a significant challenge, as oversimplification can lead to the loss of important information essential for accurate network analysis.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 250–263, 2025.
https://doi.org/10.1007/978-3-031-82439-5_21

Traditional backbone extraction methods, such as structural filtering techniques, frequently result in a substantial loss of critical information [10–14]. They often compromise the diversity of edge weights or alter the network's topology, leading to a misleading representation of the original network. To address these limitations, there is a need for more sophisticated approaches that can simplify networks while maintaining their essential characteristics.

This study proposes an innovative approach. The idea is to use link prediction techniques to build the backbone of a network [15]. Indeed, similarity-based link prediction techniques traditionally use a similarity score to estimate the likelihood of a connection between two nodes. It can be adapted to identify and retain the most significant edges in a network. Based on the similarity scores, one can determine which edges are crucial to the network's structure and which can be removed without compromising its integrity.

The study relies on global link prediction techniques. Indeed, local similarity-based methods, such as Common Neighbors and Preferential Attachment, focus on the immediate neighborhood of nodes missing broader network patterns. In contrast, global methods, such as the Katz index [16] and Shortest Path [16], consider the entire network structure. This holistic approach allows for identifying and retaining the most structurally significant edges. Applying these global techniques offers a more comprehensive backbone extraction that captures local and global network features.

To demonstrate the effectiveness of our approach, we apply it to the world air transportation network [17], a complex and highly interconnected system that exemplifies the challenges of backbone extraction. Our comparative analysis includes traditional backbone extraction methods (the Disparity filter [18] and the High Salience Skeleton filter [19]) and global similarity-based link prediction techniques (Shortest Path and Katz index). Furthermore, we examine the backbones of various real-world networks an American elementary school social network, and the Les Misérables network, focusing on basic topological properties, weight, and degree distributions. The results show that our methodology not only preserves essential structural properties, such as node reachability and edge weight distribution but also provides a more explicit representation of the network's hub-and-spoke topology—a common feature in transportation networks.

The broader implications of this research extend to various real-world applications. In global transportation networks, accurate backbone extraction can optimize routes, improve network resilience, and enhance operational efficiency. Similarly, in biological networks, our method can help maintain critical interactions while simplifying the network, leading to more accurate models of cellular processes.

The remainder of this paper is structured as follows: Sect. 2 reviews the background on backbone extraction and link prediction methodologies. Section 3 introduces the link prediction-based method for backbone extraction. Section 4 details the data and methods used in our experiments, and Sect. 5 presents the

results of our comparative analysis. Finally, Sect. 6 offers concluding remarks and outlines directions for future research.

2 Background

2.1 Backbone Extraction

Backbone extraction is a process used to simplify complex networks while maintaining their essential structure and important features. The goal is to reduce the complexity of the network to make it easier to analyze and visualize while preserving the most significant connections [20]. These methods are typically categorized into two main types: statistical and structural methods [21,22].

Statistical methods employ hypothesis testing or empirical weight distribution techniques to assess the significance of an edge. Approaches such as the Disparity filter, Polya Urn filter, Noise Corrected Filter, Marginal Likelihood filter, ECM filter, GloSS, and LANS filter illustrate this framework [23]. Hypothesis testing involves contrasting the observed edge weights against a null model to derive p-values, which indicate the degree of deviation from the null hypothesis. In contrast, empirical weight distribution methods assess edge significance by comparing the observed weights with the overall weight distribution.

Structural methods emphasize the topological characteristic of the network to extract a backbone with specific topological properties. Examples of this approach include the High Salience Skeleton filter, Metric and Ultrametric Backbone filters, H-Backbone filter, Planar Maximally Filtered Graph filter, and Primary Linkage Analysis filter [24].

2.2 Similarity-Based Link Prediction

Link prediction is a fundamental issue in network analysis [25]. It aims to infer the likelihood of a link between two nodes based on the observed network structure. Similarity-based link prediction techniques can be largely categorized into three primary categories: local, global, and quasi-local methods.

Local methods focus on the immediate neighborhood of the nodes in question. They are computationally efficient and easy to implement. Examples include Common Neighbors, Jaccard's Coefficient, and Adamic-Adar index.

Global methods consider the entire network structure when calculating the likelihood of a link. They are generally more accurate, but computationally intensive. Examples include the Katz index and the shortest path index.

Quasi-local methods serve as a compromise between local and global methods, balancing efficiency with accuracy. They extend the local methods by incorporating some global information, resulting in improved prediction performance without the extensive computation required by global methods.

3 Backbone Extraction Using Global Link Prediction Methods

This methodology is similar to statistical backbone extraction methods. These methods extract edges that exhibit significant deviations from a defined null model. However, our approach identifies edges that deviate from the predictions made by the similarity function used in a link prediction algorithm, thereby emphasizing the less predictable connections within the network. The algorithm executes the following steps:

1. For each link in a network, remove it and compute its similarity score according to a link prediction method.
2. Rank the edges in ascending order of their similarity scores
3. Select a specified fraction of top-ranked edges in the network and remove the other edges and the subsequent isolated nodes to build the backbone

3.1 Global Similarity Functions

In a preliminary work, we used local similarity-based link prediction methods, such as Common Neighbor and Preferential Attachment, to extract the backbones. In this work, we investigate global similarity functions. Specifically, we use the Shortest Path and Katz indices.

- **Shortest Path** [15]**:** The Shortest Path method is based on the idea that nodes closer in terms of the shortest path length are more likely to be linked. The similarity score is inversely related to the length of the shortest path, calculated as follows:

$$S(x, y) = -|d(x, y)| \tag{1}$$

  where $d(x, y)$ represents the shortest path length calculated using Dijkstra's algorithm [26] between the node pair (x, y).
- **Katz Index** [15]**:** The Katz method considers not only the shortest path, but also the number of all paths between a pair of nodes, exponentially damped by the length of the path to account for the fact that longer paths are less influential. The Katz index for a pair of nodes is defined as follows:

$$S(x, y) = \sum_{\ell=1}^{\infty} \beta^{\ell} \cdot |paths^{\ell}_{(x,y)}| \tag{2}$$

  Here, β is a damping factor (typically $0 < \beta < 1$) and $|paths^{\ell}_{(x,y)}|$ is the number of paths of length ℓ between nodes x and y.

4 Data and Methods

4.1 Dataset

The experiment involves three real-world networks originating from various origins (infrastructural, social, and character).

- **World Air Transportation** [17]**:** Nodes in the network represent world airports, edges denote direct flights between them, and edge weights indicate the number of flights operated by various airlines from May 17 to May 22, 2018.
- **American Elementary School** [27]**:** Nodes represent Kindergarten through 6th-grade students at a suburban elementary school in Utah (USA). Edge weights represent the total number of interactions between student pairs over two days, January 31st and February 2nd, 2013.
- **Les Misérables** [28]**:** Nodes depict actors from Victor Hugo's novel "Les Misérables," connected if they appear together in the same chapter. Edge weights indicate how many times such co-appearances occur in the novel.

Table 1 reports their basic topological properties.

Table 1. The network topological features: $|N|$, $|E|$, $< k >$, and ρ are the number of nodes, edges, average degree, and density.

Network	$\|N\|$	$\|E\|$	$< k >$	ρ
US-ES	339	16,546	97.61	0.289
World Air Transportation	2,729	16,654	12.205	0.004
Les Misérables	77	254	6.59	0.087

4.2 Backbone Evaluation Measures

Below, we specify the evaluation metrics used in the experiment.

- **Reachability** [29]**:** This metric quantifies the connectivity between node pairs in a network. It is defined as the fraction of node pairs that can communicate: $R = \frac{1}{n(n-1)} \sum_{i \neq j \in G} R_{ij}$, where n is the number of nodes, and $R_{ij} = 1$ if a path exists between nodes i and j, otherwise $R_{ij} = 0$. Reachability values range from 0 to 1, with $R = 1$ indicating full connectivity (all node pairs can communicate) and $R = 0$ indicating complete isolation (no node pairs can communicate).
- **Two-Sample Kolmogorov-Smirnov** [30]**:** The two-sample Kolmogorov-Smirnov (KS) test evaluates whether two samples follow the same distribution. The KS statistic, D, is the maximum distance between the two samples' Empirical Cumulative Distribution Functions (ECDFs).

4.3 Experimental Setup

We report two sets of experiments: First, we focus on the air transportation network. We fix the fraction of links to $\approx 10\%$. We compare classical backbone extraction methods (the Disparity filter, and the High Salience Skeleton) with the Shortest Path and Katz index similarity link prediction approach. Our goal is to qualitatively evaluate their ability to preserve essential structural characteristics such as the hub and spoke topology of the network. Second, we extract the backbones for edge fractions ranging from 0.1 to 0.9. We compute their properties, including the fraction of isolated nodes and reachability. We also compute the KS statistic between the original and the backbone's weight and degree distribution. Finally, for each network, we plot the evolution of these backbone properties as a function of the edge fraction in a series of line plots.

5 Experimental Results

5.1 Investigating the World Air Transportation Backbones with a Fixed Backbone Size

Conventional airlines often implement the hub-and-spoke strategy, where a central hub airport is the primary node for flight routes [31]. This method allows airlines to connect multiple destinations efficiently by channeling passengers through the hub, thus optimizing aircraft usage and enhancing route network connectivity. Figure 1 illustrates the World Air Transportation network with different colors representing airports by continent and node sizes corresponding to node degrees. Figures 2 and 3 display the backbones of this network comprising around 10% edges of the original network using the Disparity, High Salience Skeleton, Shortest Path, and Katz techniques.

The Disparity filter prioritizes preserving edges with much higher weights than others at the same node. This targeted retention enables the Disparity backbone to highlight connections among key continental hubs. This feature vividly illustrates essential pathways and interactions between major regions like the United States, Europe, and China. Concentrating on these crucial links, the Disparity backbone offers a clearer depiction of the main routes airplanes travel across these significant global centers.

The High Salience Skeleton filter focuses on edges (or connections) that are frequently part of the shortest paths connecting all nodes in the network. In the context of airline networks, it retains connections most often traveled when passengers move between different locations. This results in highlighting connections from local airports to regional hubs on each continent, with a few connections links between continent hubs, which exemplifies the hub-and-spoke structure employed by airlines.

The Shortest Path approach retains edges with lower similarity scores (more negative values). This implies that nodes are farther apart in the network whether considering the shortest path or a direct single link between them. Given the hub-and-spoke topology of the network, many airports have one or

Fig. 1. The World Air Transportation network with different colors representing airports by continent and node sizes corresponding to node degrees.

two connections to a regional hub. Consequently, in this network backbone, we mainly observe spoke airports primarily connected to the country's main airport or the regional hub. For example, in Argentina, all airports are connected to the main hub, Jorge Newbery Airpark (AEP) in Buenos Aires. Similarly, in South Africa, Maun Airport (MUB) and Walvis Bay Airport (WVB) are linked to Cape Town International Airport (CPT).

The prior method only considers the shortest path between the nodes connected by the edge being evaluated. In contrast, the Katz method considers all potential paths between the nodes, not just the shortest. The Katz backbone consists of edges with low Katz scores. A low Katz score indicates weak overall connectivity, which means fewer or less significant paths between nodes. Therefore, the Katz backbone overlaps by 50% with the Shortest Path backbone, maintaining nodes with single connections. The other half of the backbone includes edges with less significant paths. In the context of this transportation network, these edges connect continents through hubs. For example, the United States and Canada are connected to Australia and New Zealand. East Asia (Hong Kong, Shanghai, Tokyo, Seoul, and Taiwan) is connected to the United States airports. Lastly, the Hawaiian Islands are linked to the rest of the United States airports, which is not observed in the previous backbones.

5.2 Comparing Backbone's Topological Properties with Varying Backbone Size

Isolated Nodes Fraction: Retaining a high fraction of nodes presents a significant challenge for many techniques. This section compares various filtering

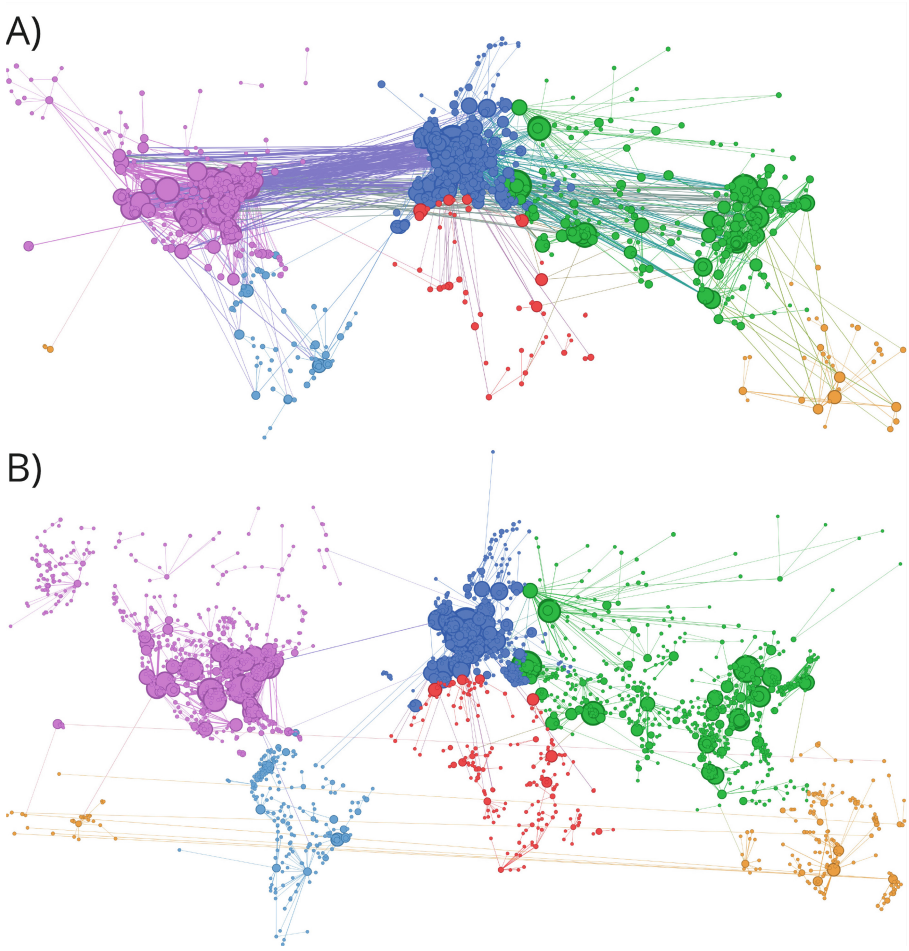


Fig. 2. The Air Transportation backbones preserving ≈ 10% of the original edges: A) Disparity backbone, and B) High Salience Skeleton backbone.

methods to determine which most effectively maintains all nodes of the original network. Figure 4 illustrates the isolated node fraction as a function of the preserved edge fraction in three different networks.

In the case of the elementary school network, represented in the middle panel, all methods manage to preserve all nodes except the SP backbone, which results in approximately 8% of the nodes being isolated at a 0.1 preserved edge fraction.

For the Air Transportation Network and Les Misérables, shown in the left and right panels, all methods except for the High Salience Skeleton (HSS) cause node isolation across all fractions. Among these, the Shortest Path (SP) method isolates the fewest nodes, followed by the Katz technique and then the Disparity Filter (DF). Conversely, the High Salience Skeleton only isolates nodes at lower

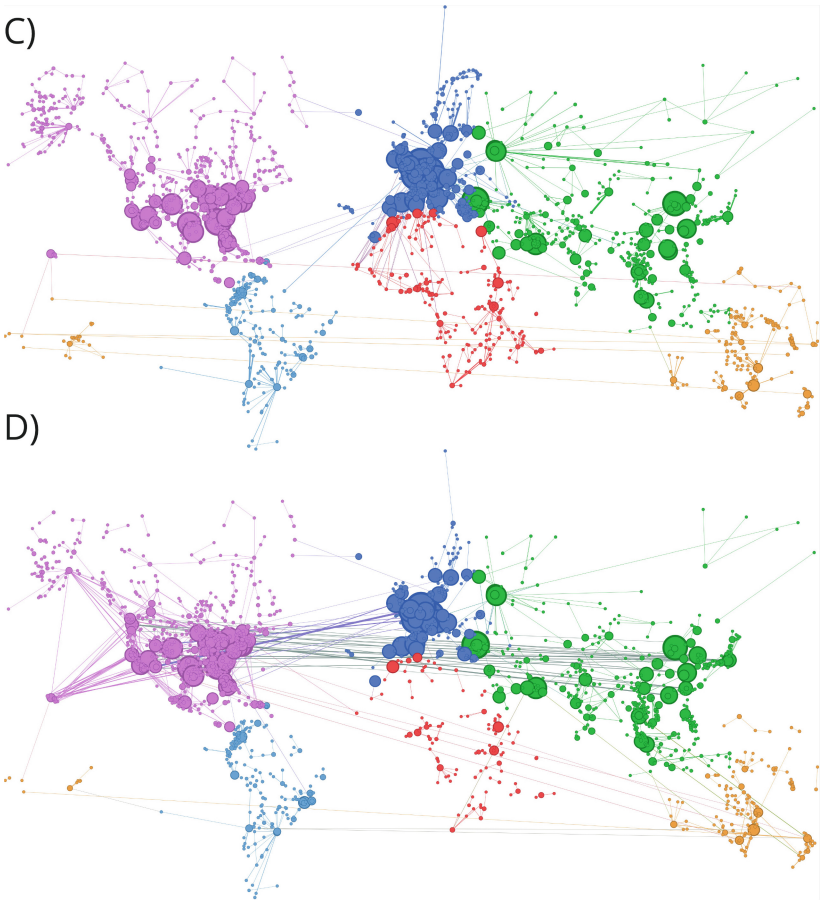


Fig. 3. The Air Transportation backbones preserving ≈ 10% of the original edges: C) SP backbone, and D) Katz backbone.

preserved edge fractions. Specifically, it isolates nodes when the backbone size is less than 20% and 30% in the World Air Transportation and Les Misérables networks, respectively.

Reachability:
Filtering the network can substantially impact its internal connectivity, potentially resulting in a backbone structure composed of multiple components. Therefore, it becomes essential to assess backbone extraction techniques by measuring the connectivity between any pair of nodes within the extracted backbones. Figure 5 demonstrates the reachability of nodes as a function of the fraction of preserved edges.

In the case of the elementary school network (middle panel), all backbones, except the Disparity backbone, maintain a reachability value of 1 across all

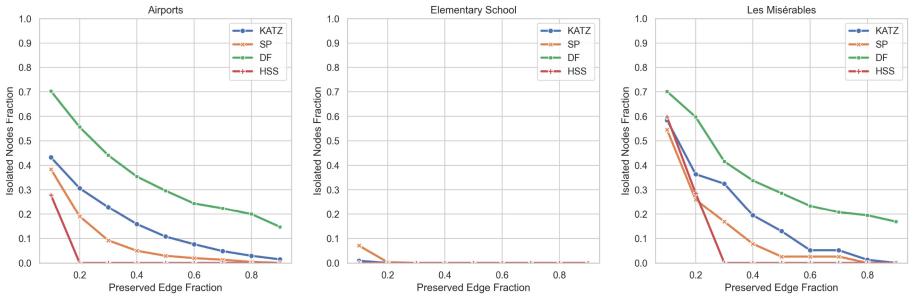


Fig. 4. Isolated node fraction as a function of the preserved edge fraction for the backbone extraction methods: Disparity Filter (DF), High Salience Skeleton (HSS), Shortest Path (SP).

edge fractions. This characteristic implies that these backbones form a single connected component regardless of the edge fraction. The Disparity backbone exhibits reduced reachability for fractions smaller than 0.3, indicating the emergence of multiple disconnected components.

Analyzing the Air Transportation Network and the Les Misérables network (left and right panels, respectively), we observe that all backbone extraction methods result in low reachability at minimal edge fractions, signifying multiple disconnected components. However, as the fraction of preserved edges increases, the High Salience Skeleton method efficiently connects the network using fewer edges than other methods, followed by the SP backbone. In contrast, the Katz and Disparity backbones necessitate a higher fraction of edges to achieve comprehensive network connectivity.

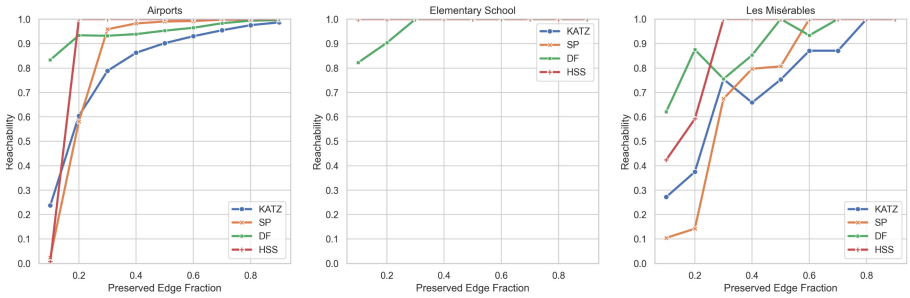


Fig. 5. Node reachability as a function of the preserved edge fraction for the backbone extraction methods: Disparity Filter (DF), High Salience Skeleton (HSS), Shortest Path (SP).

Weight Distribution: The weight scales in a network indicate the strength of connections and relationships between nodes. An optimal filtering technique

should, therefore, be capable of preserving these weights across different scales to ensure that the original network's weight distribution remains intact. Evaluating various filtering methods based on their ability to maintain the weight distribution of the original network can help identify the most effective techniques.

Figure 6 presents the Kolmogorov-Smirnov (KS) statistic as a function of the preserved edge fraction. This figure allows for a comparison between the original network's weight distribution and the weight distributions of various backbones obtained through different filtering methods.

Across all panels, it is clear that the Katz backbone exhibits the most similar weight distribution to the original network, as indicated by its lower KS statistic. Conversely, other backbones display higher KS statistics, indicating more significant deviations from the original weight distribution. Notably, the weight distributions of the Shortest Path (SP) and Disparity backbones demonstrate the greatest divergence from the original network's weight distribution. In contrast, the High Salience Skeleton backbone performs moderately between these extremes, maintaining a weight distribution that is neither too close nor too far from the original compared to the other methods.

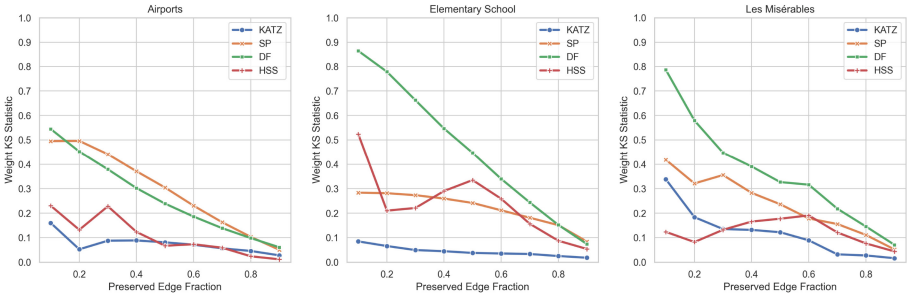


Fig. 6. The Kolmogorov-Smirnov (KS) static between the original and backbone weight distributions as a function of the preserved edge fraction for the backbone extraction methods: Disparity Filter (DF), High Salience Skeleton (HSS), Shortest Path (SP).

Degree Distribution: The degrees of nodes in a network are indicative of the connections and influence of these nodes within the network. An effective filtering method retains a diverse range of degrees, thus preserving the original degree distribution. This preservation is crucial as it maintains the network's inherent structural characteristics and ensures a representative analysis.

Figure 7 shows the KS statistic as a function of the preserved edge fraction, comparing the degree distributions of the original network with those obtained from different backbone extraction methods.

The results illustrated in the figure demonstrate that the Shortest Path (SP) and Disparity backbones exhibit the most similar degree distribution to the original network, as evidenced by their lower KS statistics across all panels. Notably,

the SP backbone achieves the lowest KS statistic in the Elementary School network (middle panel). In contrast, the Disparity backbone attains the lowest KS statistic in the Les Misérables network (right panel). In the World Air Transportation network (left panel), both the SP and Disparity backbones demonstrate overlapping and favorable KS statistics. Conversely, the High Salience Skeleton backbone shows the greatest deviation from the original network's degree distribution, indicating a significant alteration in the network's structural properties. The Katz backbone exhibits similar behavior in the Elementary School network, manifesting a degree distribution significantly different from the original. However, in the World Air Transportation and Les Misérables networks, the Katz backbone performs moderately, maintaining a degree distribution that is neither too close nor too far from the original compared to the other filtering methods.

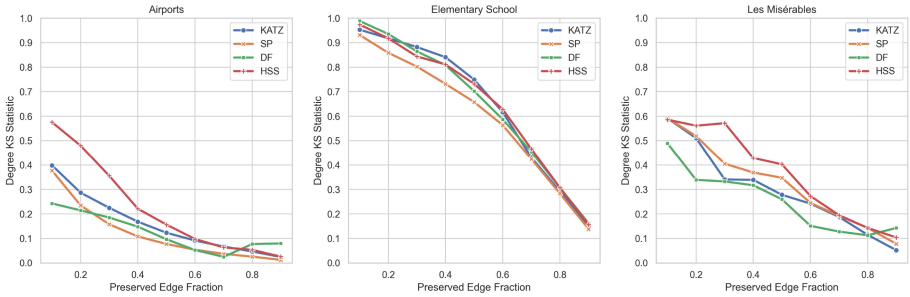


Fig. 7. The Kolmogorov-Smirnov (KS) static between the original and backbone degree distributions as a function of the preserved edge fraction for the backbone extraction methods: Disparity Filter (DF), High Salience Skeleton (HSS), Shortest Path (SP).

6 Conclusion

This study investigates global similarity-based link prediction methods for backbone extraction in complex networks. Our research demonstrates that using the scores of the link prediction algorithms enables the identification of the most significant edges.

A comparative analysis with traditional backbone extraction methods reveals differing parts of the hub-and-spoke business model used by airline companies. Specifically, the Disparity filter primarily identifies links between continental hubs, while the Shortest Path (SP) method uncovers connections between local airports and regional hubs. The Katz backbone method synthesizes these approaches by revealing both connections between local airports and regional hubs as well as some links between continental hubs.

An analysis of the properties indicates that no single method excels across all aspects. The SP and Katz backbones are more effective at maintaining the

network nodes than the Disparity backbone. However, they do not surpass the High Salience Skeleton. Conversely, the SP backbone and High Salience Skeleton outperform the others in maintaining node reachability. Additionally, the Katz backbone excels at preserving the weight distribution, whereas the SP and Disparity backbones are more effective at maintaining the degree distribution. Future research will broaden the exploration of alternative link prediction approaches for backbone extraction beyond those based on similarity.

Acknowledgment. This material is based upon work supported by the Agence Nationale de Recherche under grant ANR-20-CE23-0002.

References

1. Zignani, M., Quadri, C., Gaito, S., Cherifi, H., Rossi, G.P.: The footprints of a mastodon: how a decentralized architecture influences online social relationships. In: IEEE INFOCOM 2019, pp. 472–477. IEEE (2019)
2. Erum, I., Malick, R.A.S., Ahmed, G., Cherifi, H.: Complex network analysis of mass violation, specifically mass killing. Entropy **24**(8), 1017 (2022)
3. Güell, O., Sagués, F., Serrano, M.Á.: Detecting the significant flux backbone of *Escherichia coli* metabolism. FEBS Lett. **591**(10) (2017)
4. Kumar, P., Verma, P., Singh, A., Cherifi, H.: Choosing optimal seed nodes in competitive contagion. Front. Big Data **2**, 16 (2019)
5. Chakraborty, D., Singh, A., Cherifi, H.: Immunization strategies based on the overlapping nodes in networks with community structure. In: Nguyen, H.T.T., Snasel, V. (eds.) CSoNet 2016. LNCS, vol. 9795, pp. 62–73. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42345-6_6
6. Sow, A., Diallo, C., Cherifi, H.: Effects of vertical transmission and human contact on zika dynamics. Complexity **2022**(1), 5366395 (2022)
7. Sow, A., Diallo, C., Cherifi, H.: Interplay between vaccines and treatment for dengue control: An epidemic model. PLoS ONE **19**(1), e0295025 (2024)
8. Tilly, S., Livan, G.: Macroeconomic forecasting with statistically validated knowledge graphs. Expert Syst. Appl. **186**, 115765 (2021)
9. Diop, I.M., Cherifi, C., Diallo, C., Cherifi, H.: Revealing the component structure of the world air transportation network. Appl. Netw. Sci. **6**(1), 1–50 (2021). https://doi.org/10.1007/s41109-021-00430-2
10. Neal, Z.P.: backbone: an R package to extract network backbones. PloS One **17**(5), e0269137 (2022)
11. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: Extracting backbones in weighted modular complex networks. Sci. Rep. **11**, 12 (2021)
12. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: Extracting modular-based backbones in weighted networks. Inf. Sci. **576**, 454–474 (2021)
13. Hmaida, S., Cherifi, H., El Hassouni, M.: A multilevel backbone extraction framework. Appl. Netw. Sci. **9**(1), 41 (2024)
14. Ghalmane, Z., Cherifi, C., Cherifi, H., El Hassouni, M.: A community-aware backbone extractor for weighted networks. In: 9th International Conference on Complex Networks and Their Applications, p. p–3 (2020)
15. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: Link prediction techniques, applications, and performance: a survey. Phys. A Stat. Mech. Appl. **553**, 124289 (2020)

16. Lü, L., Zhou, T.: Link prediction in weighted networks: the role of weak ties. Europhys. Lett. **89**(1), 18001 (2010)
17. Alves, L., Aleta, A., Rodrigues, F., Moreno, Y., Amaral, L.: Centrality anomalies in complex networks as a result of model over-simplification. New J. Phys. **22**, 01 (2020)
18. Serrano, M.A., Boguna, M., Vespignani, A.: Extracting the multiscale backbone of complex weighted networks. Proc. Nat. Acad. Sci. **106**, 6483–6488 (2009)
19. Grady, D., Thiemann, C., Brockmann, D.: Robust classification of salient links in complex networks. Nat. Commun. **3**(1), 864 (2012)
20. Kirkley, A.: Fast nonparametric inference of network backbones for graph sparsification (2024)
21. Yassin, A., Cherifi, H., Seba, H., Togni, O.: A modular network exploration of backbone extraction techniques. In: International Conference on Complex Networks and Their Applications, pp. 296–308 (2023)
22. Yassin, A., Haidar, A., Cherifi, H., Seba, H., Togni, O.: An evaluation tool for backbone extraction techniques in weighted complex networks. Sci. Rep. **13**(1), 17000 (2023)
23. Yassin, A., Cherifi, H., Seba, H., Togni, O.: Air transport network: a comparison of statistical backbone filtering techniques. In: International Conference on Complex Networks and Their Applications, pp. 551–564. Springer, Cham (2022)
24. Yassin, A., Cherifi, H., Seba, H., Togni, O.: Air transportation network backbone extraction: a comparative analysis of structural filtering techniques. In: Hà, M.H., Zhu, X., Thai, M.T. (eds.) Computational Data and Social Networks, pp. 337–348. Springer, Singapore (2024)
25. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: Link prediction techniques, applications, and performance: a survey. Phys. A Stat. Mech. Appl. **553**, 124289 (2020)
26. Dijkstra, E.W.: A note on two problems in Connexion with graphs. Numer. Math. **1**, 269–271 (1959)
27. Stehlé, J., et al.: Simulation of an seir infectious disease model on the dynamic contact network of conference attendees. BMC Med. **9**, 87 (2011)
28. Knuth, D.: The Stanford GraphBase. A Platform for Combinatorial Computing (1993)
29. Sato, Y., Ata, S., Oka, I.: A strategic approach for re-organization of internet topology for improving both efficiency and attack tolerance (2008)
30. Hodges, J.L.: The significance probability of the Smirnov two-sample test. Ark. Mat. **3**(5), 469–486 (1958)
31. Yassin, A., Cherifi, H., Seba, H., Togni, O.: Exploring statistical backbone filtering techniques in the air transportation network. In: 2022 IEEE Workshop on Complexity in Engineering (COMPENG), pp. 1–8 (2022)

Mobility and Networks

Kernel-Based Recovery of Traffic Graph Signals on Urban Networks

Rafael Alejandro Martínez Márquez[1,2](✉) and Giuseppe Patanè[1]

[1] CNR-IMATI "E. Magenes",
Via De Marini 6 - Torre Francia, 16149 Genoa, Italy
{rafael.martinez,patane}@ge.imati.cnr.it
[2] Department of Mathematics, University of Genoa, Via Dodecaneso 35, 16146 Genoa, Italy

Abstract. Acquisition of traffic data in an urban network is performed using a variety of sources, however, each of them has a coverage limit due to distinct reasons, such as privacy concerns or implementation costs. Since the recorded information is contained in a non-Euclidean structure and is partially unknown, we propose modelling an urban network as a road graph (i.e. with roads as nodes, and edges representing their intersections) and traffic data as incomplete graph signals to be recovered via kernel ridge regression with a kernel matrix aware of the graph topology. Evaluating distinct graph kernels based on the Laplacian matrix to recover graph signals constructed from a real-world traffic dataset (i.e., the PNeuma dataset for Athens), we show that the Laplacian RBF kernel obtains the best results in terms of smoothness and that the vehicle volume signal is well recovered with any kernel.

Keywords: Complex Networks and Mobility · Graph Signal Processing · Kernel Ridge Regression · Traffic Data Recovery · Urban Network · Road Graph

1 Introduction

Estimating road traffic states in an urban network is crucial for intelligent transportation systems and traffic management. Moreover, a precise estimation requires a large amount of data over the area of interest, however, the collecting methods (e.g. sensors, drones, monitoring stations) have a capacity limit for data acquisition due to physical conditions, privacy concerns, or implementation costs [7,10]. Such traffic data is collected on irregular domains and can be modelled using graph-based approaches, being the graph signal recovery methods [4,8,14] a suitable option to estimate traffic state values on a whole urban network when they are available only on a small subset of roads.

Usual approaches are based on the assumption that the smoothness of the recovered signal can be quantified through a positive function involving an ℓ_p-norm [3] or the Laplacian quadratic form [8]. These methods are based on the

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 267–278, 2025.
https://doi.org/10.1007/978-3-031-82439-5_22

trade-off between the precision of the signal values on the sampled nodes, measured through the empirical error, and the quantification of the graph signal smoothness. The recovered signal corresponds to the solution of a minimisation problem involving both values, which is governed by a regularisation parameter, which controls the effect of both quantities on the recovered signal. A more general approach consists of assuming that a graph signal is contained in a *Reproducing Kernel Hilbert Space (RKHS)* [17] and the smoothness is quantified by its associated norm, which is determined by a continuous function composed with the quadratic form associated to a symmetric positive semi-definite matrix called the *kernel matrix*. When such a continuous function is the quadratic function and the empirical error is measured through the square of the ℓ_2-norm, the solution of the minimisation problem is called the *kernel ridge regression*. Well-known kernels include *diffusion kernels* [11], and kernels obtained from a spectral filter, which is a smooth real-valued function evaluated on the graph Laplacian or its normalised version (e.g. *random walk kernels* and *Laplacian regularisation kernels*) [15].

This paper addresses the recovery of traffic data (e.g. vehicle volume, average speed) on an urban network via kernel ridge regression with different kernel matrices, by modelling it as a graph signal in a RKHS. We start reviewing the main variables (e.g. volume, density, speed) used in Traffic Flow Theory [9] (Sect. 2). By understanding these foundational aspects, we can better appreciate their implications for traffic analysis and modelling. Moreover, we present kernel methods approaches that have been applied for the estimation of the distribution of traffic variables from a sample of trajectory data, as well as the main notions of graph signals in RKHS and the kernel-based graph signal recovery problem that leads to the definition of the kernel ridge regression, which constitutes a robust method to obtain consistent traffic information when the traffic variables are modelled using a network-based structure.

Then (Sect. 3), we generate graph signals on the road graph associated with an urban network (i.e. each node represents a road and each edge represents the intersection between two roads) using a real-world traffic dataset. Such graph signals were generated from the PNeuma data set [1], i.e. a real-world traffic data set available in Athens. We also design various kernels that take into account the topology of the graph and use them to recover entirely the traffic graph signals, which were only available on a small subset of roads. Furthermore, we evaluate the impact of the trade-off between the empirical error on the sampled nodes and the kernel-based smoothness induced by the RKHS norm on the recovered signals by systematically modifying the regularisation parameter. This parameter can also be leveraged to find the kernel that is the most suitable to recover traffic signals in terms of small empirical error and high smoothness, and also visualise the recovered signals when using distinct kernels with a fixed regularisation term. Finally, we present the main conclusions and future research directions (Sect. 4).

2 Related Work

Traffic Flow Theory and Kernel Density Estimation. Historically, the definition of traffic variables and their associated numerical results have been related to the methods of measurement. Earliest traffic investigators had at their disposal only manual counters, where collected information is expressed as averages of observed data at a point. Many operational decisions are still influenced by point measurements even at present [9]. Later-developed methods include photographs or videos that facilitate the computation of scaled values, and cross-section units that register information of vehicles crossing them. The *trajectory* of a vehicle describes its position over time along a road segment, and it is the core variable in traffic flow theory [18]. The superposition of families of vehicles' trajectories with distinct characteristics (e.g. fast or slow drivers) describes *stationary traffic* if the observed variables do not tell any particular information, e.g. time of the day, weather conditions, or road disruptions, by inspecting the time-space diagram, i.e. a two-dimensional representation of the vehicles' positions at different times [6]. While stationary analyses are valuable for design purposes, more in-depth studies of operations often need an understanding of traffic dynamics.

Vehicle trajectory measurements usually contain information about the average characteristics of the traffic flow at the microscopic level (e.g. instant speed of a vehicle). In contrast, at the macroscopic level traffic flow variables (e.g. volume, density, and speed) reflect the average state of traffic in contrast to the microscopic variables, which focus on individual drivers. Volume is often defined as the average number of vehicles that pass a cross-section during a unit of time. Density is typically defined as the number of vehicles per distance unit. Similarly, average speed can be defined at a cross-section (local mean speed) or at a time instant (instantaneous mean speed). The *fundamental diagrams* describe the relation between the macroscopic traffic variables. They are built upon the premise that under similar traffic conditions, drivers will behave similarly, and upon the *continuity equation*, which states that volume is equal to density times speed under ideal conditions. Several studies concerning the definition of the traffic flow variables when using distinct data collection methods, the fundamental relationship between them, and more advanced models, have been carried out [9,13,18].

The incomplete collection of trajectory data in urban networks drives efforts to estimate the distribution of traffic variables at unknown points by modelling them as random variables, taking into account the influence of the unknown underlying distribution of the observed data. *Kernel density estimation* (KDE) is one of the most well-known approaches to estimating the underlying probability density function of a data set [5]. It has been applied in the traffic modelling domain for traffic density estimation in Intelligent Transport Systems [16], traffic speed estimation via kernel support vector regression [19], and for the modelling via Gaussian kernels of the distribution of vehicle headways [20], which is another important traffic flow measure defined as the time between consecutive vehicle observations at a fixed location. Formally, KDE consists of estimating the density of a random variable when a finite set of independent and identically

distributed samples with unknown distribution are available. Instead of assuming a specific parametric distribution, KDE estimates the density of a random variable by placing a smooth function (i.e., the kernel function) at each observed data point. The overall density is then obtained as the average of these kernel functions, scaled by a parameter. Each kernel function is centred at an observed data point and spreads its influence over nearby values, thus creating a smooth approximation of the probability density function. The existent approaches for estimating distributions of traffic variables via kernel methods are placing the kernel functions at each of the vectors obtained through trajectory data without considering any further structural or topological dependencies, as typically done in network-based models. Such approaches focus solely on the distribution of the data points themselves, rather than on their relationships or interactions within a network. In addition, network-based models offer a powerful framework for complementary analysis, especially in identifying key nodes based on their connectivity. This is particularly valuable for geographically embedded systems like urban networks, where understanding the flow and structure of connections can reveal hidden dynamics [12].

Kernel Ridge Regression for Graph Signals. Consider a graph $G = (\mathcal{V}, \mathbf{W})$ where $\mathcal{V} = \{1, \cdots, N\}$ is the set of nodes, and $\mathbf{W}$ is the adjacency matrix with non-negative entries. An edge (i, j) exists if and only if the corresponding weight in the adjacency matrix satisfies $W_{ij} > 0$. The *degree matrix* $\mathbf{D}$ is a diagonal matrix with entries $D_{ii} = \sum_{j=1}^{N} W_{ij}$, and the *combinatorial graph Laplacian* of an undirected graph is defined through $\mathbf{L} = \mathbf{D} - \mathbf{W}$. A *graph signal* $\mathbf{x}\colon \mathcal{V} \longrightarrow \mathbb{R}$ is a mapping of the graph nodes to the real numbers, i.e., to each node $i \in \mathcal{V}$ it is assigned a value $x_i \in \mathbb{R}$. Assuming that graph signals belong to a RKHS, i.e. a Hilbert space

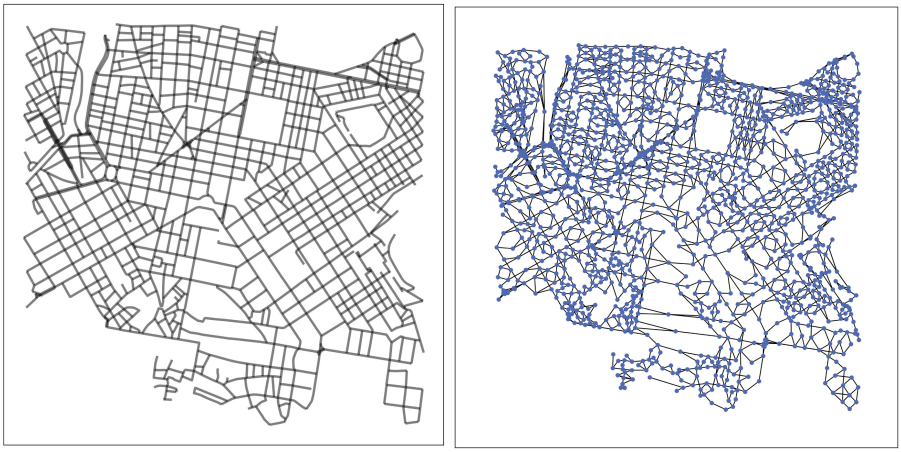
$$\mathcal{H} = \left\{\mathbf{x} \in \mathbb{R}^N \colon \mathbf{x} = \mathbf{K}\boldsymbol{\alpha} \text{ for some } \boldsymbol{\alpha} \in \mathbb{R}^N\right\},$$

defined by a symmetric and positive semidefinite matrix $\mathbf{K}$ called the *kernel matrix*, we can measure their smoothness by computing the value $s = \Omega(\|\mathbf{x}\|_{\mathcal{H}})$, i.e. a real valued increasing function composed with the *RKHS semi-norm* of $\mathbf{x}$ that is given by $\|\mathbf{x}\|_{\mathcal{H}} = \left(\boldsymbol{\alpha}^\top \mathbf{K}\boldsymbol{\alpha}\right)^{1/2}$. Smaller values of s stand for *smoother* graph signals.

Given a noisy sample $\mathbf{y}_{\mathcal{S}}$ of a signal on a subset $\mathcal{S} \subseteq \mathcal{V}$, the *kernel-based* graph signal recovery problem consists of estimating the original signal on every node through solving the optimisation problem

$$\mathbf{x}^* = K\left(\underset{\boldsymbol{\alpha}\in\mathbb{R}^N}{\arg\min}\, \mathcal{L}(\mathcal{S}, \mathbf{y}_{\mathcal{S}}, (\mathbf{K}\boldsymbol{\alpha})_{\mathcal{S}}) + \lambda\Omega\left(\left(\boldsymbol{\alpha}^\top \mathbf{K}\boldsymbol{\alpha}\right)^{1/2}\right)\right).$$

According to [14] the recovered graph signal can be expressed as $\mathbf{x}^* = \mathbf{K}\mathbf{C}^\top\boldsymbol{\beta}$ for some $\boldsymbol{\beta} \in \mathbb{R}^m$, where $\mathbf{C} \in \mathbb{R}^{m\times N}$ is the *sampling matrix* with entries in $\{0, 1\}$ that satisfies $\mathbf{C}\mathbf{x}^* = (\mathbf{K}\boldsymbol{\alpha})_{\mathcal{S}}$. When the matrix $\mathbf{C}\mathbf{K}\mathbf{C}^\top \in \mathbb{R}^{m\times m}$ is full rank, $\mathcal{L}(\mathcal{S}, \mathbf{y}_{\mathcal{S}}, \mathbf{x}) = \|\mathbf{y}_{\mathcal{S}} - \mathbf{x}_{\mathcal{S}}\|_2^2$, and $\Omega(\zeta) = \zeta^2$, the kernel-based recovered graph signal is called the *kernel ridge regression* and is given by



a) Athens' urban network b) Athens' road graph

Fig. 1. Urban network of Athens generated by OSMnx a) and its associated road graph b) where each of the 1941 nodes represents a road, and an edge represents the junction between two roads.

$$\mathbf{x}^* = \mathbf{K}\mathbf{C}^\top \left(\mathbf{C}\mathbf{K}\mathbf{C}^\top + \lambda m \mathbf{I}\right)^{-1} \mathbf{y}_{\mathcal{S}},$$

where $\mathbf{I}$ is the identity matrix in $\mathbb{R}^{m \times m}$. In the context of kernel methods, kernel ridge regression for graph signals provides a way to estimate missing data by focusing not only on the data points themselves, but also on their interactions within the network structure.

3 Traffic Graph Signals and Kernel-Based Recovery

We apply kernel-based methods to the recovery of traffic data on an urban network which has been acquired only on a subset of roads of the network. To this end, we generate distinct graph signals from a real-world traffic dataset on a subset of nodes of the Athens road graph, and design graph kernels that can recover them on the whole network via kernel ridge regression. We additionally perform a trade-off analysis between the empirical error on the sampled nodes, measured through the normalised maximum error, and the kernel-based smoothness of the recovered signal determined by the RKHS norm.

Traffic Graph Signals on Road Graphs in Urban Networks. Given an urban network of a city with N roads, we define its associated *road graph* as the graph $G = (\mathcal{V}, \mathbf{W})$ with N nodes where each node represents a road, and an edge represents an intersection between 2 roads. We generated the urban network of Athens (Fig. 1a) using the OSMnx Python library [2] and constructed its associated road graph (Fig. 1b). The entries of $\mathbf{W}$ are defined as the inverse

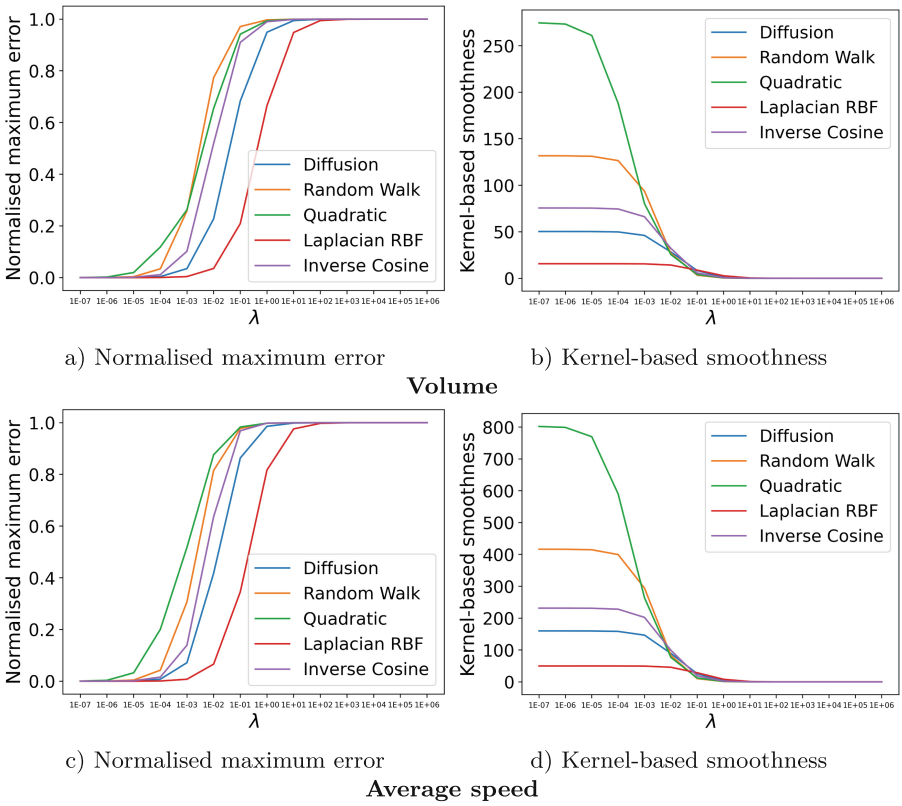


Fig. 2. Normalised maximum error on the sampled nodes for various kernels (a),c)), and the corresponding kernel-based smoothness of the recovered graph signal (b),d)).

of the distance (in meters) along the trajectories of 2 linked roads when connecting their associated midpoints, i.e. $w_{ij} = \left(\frac{\ell_i}{2} + \frac{\ell_j}{2}\right)^{-1}$, where $\ell_i, \ell_j > 0$ are the length of the roads i and j, respectively, which are provided by OSMnx. The choice of the edges' weights means that two adjacent short roads will be strongly connected while two adjacent long roads will have a weak connection. The obtained Athens road graph is connected and undirected and consists of 1941 nodes and 3400 edges. To define graph signals from real traffic data on the nodes of the Athens' road graph, we used the pNeuma data set [1], which consists of massive multi-modal data collected by ten drones hovering over the central business district of Athens that recorded multi-modal traffic streams in the central business district of Athens. We fixed a time instant and used the "Car" category vehicles to construct the following graph signals on the nodes of the Athens road graph:

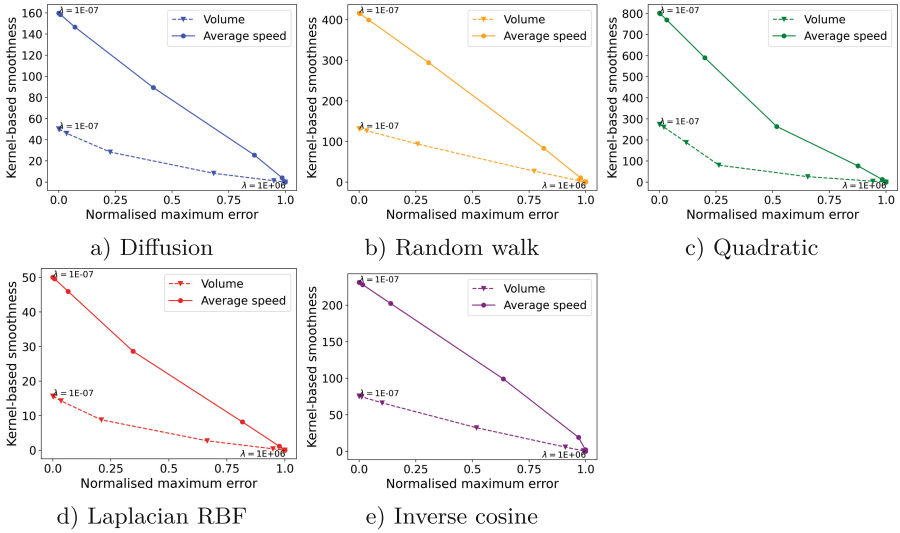


Fig. 3. Combined analysis of the normalised maximum error and the kernel-based smoothness for the volume and average speed signals when modifying systematically the regularisation parameter λ.

1. The *volume* signal, i.e. the number of vehicles whose geometrical projection of its geographical position to the road graph belongs to the associated road.
2. The *average speed* signal, i.e. the arithmetic mean of the instant speed (km/h) of the vehicles projected to the associated road.
3. The *density* signal, i.e. the quotient between the volume signal value and the length of the road.

Graph Kernels Design. The exponentiation operation on matrices provides naturally the positive semi-definite property for kernels, in fact, [11] shows that matrices of the form $\mathbf{K} = e^{\beta \mathbf{H}}$ are symmetric and positive semi-definite for a real parameter β when $\mathbf{H}$ is a symmetric matrix, and consequently, they are potential choices for a kernel matrix, called *diffusion kernels*. Since the Laplacian matrix $\mathbf{L}$ and its normalised version $\widetilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ are symmetric matrices and, motivated by the demand of implementing kernels that capture the graph topology [14], we designed a diffusion kernel $\mathbf{K} = e^{2\mathbf{L}}$. The spectral decomposition $\widetilde{\mathbf{L}} = \mathbf{U}\mathrm{diag}\left(\lambda_1, \ldots, \lambda_N\right)\mathbf{U}^\top$ of the normalised Laplacian allows us to design additional graph kernels by applying a real-valued smooth function $h(r)$ to the eigenvalues $\lambda_1, \ldots, \lambda_N$ of $\widetilde{\mathbf{L}}$ (*spectral frequencies*) and get a kernel $\mathbf{K} = \mathbf{U}\mathrm{diag}\left(h(\lambda_1), \ldots, h(\lambda_N)\right)\mathbf{U}^\top$ generated by a spectral filter. In this fashion, we also obtain the *random walk kernel* generated by the function $h(r) = (3-r)^{-1}$, the *inverse cosine* kernel $h(r) = (\cos \pi r/4)^{-1}$, the *Laplacian Radial Basis Function (RBF) kernel* obtained from the function $h(r) = 1 + 100r$ [15], and the quadratic kernel induced by $h(r) = r^2$ to test and compare their performance when recovering a traffic graph signal.

Traffic Signal Recovery on Athens' Roads: Experimental Results. For a fixed time instant, the position of vehicles within the "Car" category recorded during the PNeuma experiment [1] was projected to the roads of Athens, leading to graph signals supported on 172 out of the total 1941 roads in the Athens' road graph; we consider these roads as the sampled nodes $\mathcal{S}$ used to recover the signal values on the rest of the network. For each graph Laplacian kernel, to measure the empirical error we compute the *normalised maximum error* defined as $\varepsilon = \|\mathbf{x}_{\mathcal{S}} - \mathbf{C}\mathbf{x}^*\|_\infty / \|\mathbf{x}_{\mathcal{S}}\|_\infty$ on the sample nodes $\mathcal{S}$ when recovering the signal through kernel ridge regression, where $\mathbf{x}^*$ is the recovered signal from the sample $\mathbf{x}_{\mathcal{S}}$, and $\mathbf{C} \in \mathbb{R}^{m \times N}$ is the sampling operator on $\mathcal{S}$, i.e. being $\mathcal{S} = (v_{\pi(1)}, \ldots, v_{\pi(m)})$ then $C_{ij} = 1$ if $j = \pi(i)$ and 0, otherwise. The lower the normalised maximum error is, the smaller the difference between the recovered and ground-truth signals' values will be. To study the behaviour of the normalised maximum error for a given kernel we modify systematically the regularisation value of λ by parametrising it as $\lambda = 10^a$ for $a = -7, -6, \ldots, 5, 6$, and evaluate the empirical error of its corresponding kernel ridge regression. Furthermore, for the same values of λ we evaluate the RKHS norm for the recovered signal using each of the listed kernels.

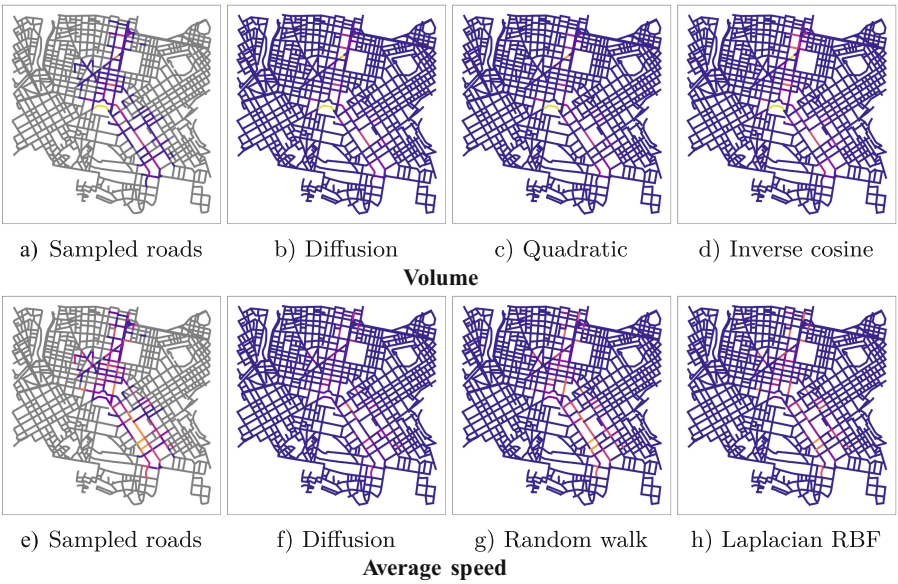


Fig. 4. Volume signal values available on 172 roads considered as sample nodes a), and the recovered graph signal on the 1941 roads of the Athens' road graph when using diffusion b), quadratic c), and inverse cosine d) kernels with parameter $\lambda = 2$. Sample of the average speed signal e), and its recovered version using diffusion f), random walk g) and Laplacian RBF h) kernels with parameter $\lambda = 2$.

For the volume signal, the normalised maximum error (Fig. 2a) shows a similar trend for all of the chosen graph kernels, i.e. it increases when the parameter λ grows. In particular, it is converging to 1 when λ is exceeding the order of magnitude of 2, since the increase of λ causes the minimisation of the kernel-based smoothness and therefore the increase of the empirical error, which in its normalised version is bounded by 1. The Laplacian RBF kernel is the one that gives the smallest normalised maximum error, indeed it obtains the same error values with λ being at least one order of magnitude greater than with the other kernels (e.g. between the orders of -3 to 1). In contrast, when the parameter λ is decreasing to orders of magnitude below -6, the empirical error on the sampled nodes is getting closer to zero for each of the kernels. Similarly, the behaviour of the kernel-based smoothness (Fig. 2b), measured through the RKHS norm of the recovered signal, is monotone decreasing in contrast with the trend of the normalised maximum error for each of the kernels. The Laplacian RBF kernel is the one with the smallest kernel-based smoothness, and the other kernels attain the same smoothness values only for the positive orders of magnitude. This means that for each of the considered kernels, the optimisation problem prioritises the smoothness rather than the precision on the sample nodes when the order of magnitude of the regularisation parameter λ is positive.

For the average speed signal, the normalised maximum error (Fig. 2c) also shows an increasing trend as with the volume signal, with a particular remark: the quadratic kernel gives the largest of the normalised maximum errors for each of the considered values of λ, in contrast with the case of the volume signal where the random walk kernels give the largest values of the error when λ is of an order of magnitude greater than -3. The observed behaviour of the kernel-based smoothness (Fig. 2d) is almost identical to the case of the volume signal, being the only significant remark that the values for the average speed signal are larger (roughly three times the ones for the volume signal) for each of the considered kernels.

A combined analysis of the normalised maximum error and the kernel-based smoothness (Fig. 3) shows that for each of the considered kernels, the kernel ridge regression works better for the volume signal than for the average signal because for the same empirical error values the recovered volume signal is smoother than the recovered average speed signal. Also, the range of the kernel-based smoothness (y axis) shows that the Laplacian RBF kernel (Fig. 3d) is the most suitable to recover the traffic signals by kernel ridge regression since the smoothness values are smaller for either the volume and the average speed signals. Following the same reasoning, the second most suitable kernel is the diffusion one (Fig. 3a) while the most unfavourable one is the quadratic (Fig. 3c). It is also remarkable to mention that for each kernel and graph signal the normalised maximum error is converging to 1, which shows the consistency of the methods.

In summary, the results show that it is possible to use kernel ridge regression for the recovery of traffic-induced signals without compromising the precision of the values on the sampled nodes in pursuit of smoothness for the recovered signal. Moreover, the possibility of writing a kernel as a spectral filter allows us to

control the effect of each of the spectral frequencies by choosing an appropriate smooth generator function. Also, it is relevant to mention that regardless of the differences in the kernel-based smoothness of the various graph kernels, the distribution of the recovered signal is consistent and preserves the possibility of distinguishing the transition between low and high values either for the volume signal (Figs. 4a-d)) and for the average speed signal (Figs. 4e-h)).

4 Conclusions and Future Work

In this paper, we examined a variety of graph kernels that can accomplish the task of recovering a traffic-induced graph signal on an urban network through the kernel ridge regression method. The study of the trade-off between the empirical error on the sampled nodes and the RKHS norm of the recovered signal shows that the kernel ridge regression method can recover traffic-induced graph signals with a controlled balance between the error on the sampled nodes and the kernel-based smoothness of the recovered signal, for distinct topology-aware graph kernels based on the Laplacian matrix. The results also show that for the volume signal, any of the considered kernels can recover it as a highly smooth signal with a small empirical error. Also, the Laplacian RBF kernel was the most favourable one since the ranges of the normalised maximum error and the kernel-based smoothness of the recovered signals are considerably small when modifying the regularisation parameter. Large traffic data sets such as the one obtained through the PNeuma experiment provide the possibility to generate graph signals of different natures on the road graph associated with an urban network. This exhibits the potential of kernel-based graph signal recovery methods (e.g. kernel ridge regression) to recover the value of traffic variables on the whole network from only a small subset of known values.

While the kernel ridge regression method was suitably tested on the PNeuma data set with traffic signals constructed from trajectories of vehicles in Athens, it is essential to perform additional tests on distinct urban networks to assess its ability to be generalised to other data sets with different characteristics, and networks with diverse topological complexities, and it constitutes part of possible extensions of this work. Moreover, the use of directed edges must be explored, since undirected edges do not account for directional traffic flows, which is critical in real-world traffic networks. In addition, it is relevant to the implementation of different edge weights to evaluate their impact on the recovery problem, and the incorporation of geographical features and external factors that may affect the estimation of traffic states in an urban network. Finally, the examination of relationships between the recovered traffic graph signals (e.g. through fundamental diagrams) could enhance the analysis of traffic states in an urban network.

Acknowledgements. Our sincere recognition to the anonymous reviewers, whose constructive feedback has enhanced the quality of this manuscript.

Funding Information. This research was partially funded by PON "Ricerca e Innovazione" 2014-2020, Asse IV "Istruzione e ricerca per il recupero", Azione IV.5 "Dot-

torati su tematiche Green" DM 1061/2021 - FSE REACT-EU. G. Patanè has been partially supported by the European Commission, NextGenerationEU, Missione 4 Componente 2, "Dalla ricerca all'impresa", Innovation Ecosystem RAISE "Robotics and AI for Socio-economic Empowerment", ECS00000035.

References

1. Barmpounakis, E., Geroliminis, N.: On the new era of urban traffic monitoring with massive drone data: the pneuma large-scale field experiment. Transp. Res. Part C: Emerg. Technol. **111**, 50–71 (2020)
2. Boeing, G.: Osmnx: new methods for acquiring, constructing, analyzing, and visualizing complex street networks. Comput. Environ. Urban Syst. **65**, 126–139 (2017)
3. Chen, S., Sandryhaila, A., Moura, J.M.F., Kovačević, J.: Signal recovery on graphs: variation minimization. IEEE Trans. Sig. Process. **63**(17), 4609–4624 (2015)
4. Chen, S., Varma, R., Singh, A., Kovačević, J.: Signal recovery on graphs: fundamental limits of sampling strategies. IEEE Trans. Sig. Inf. Process. Netw. **2**(4), 539–554 (2016)
5. Chen, Y.C.: A tutorial on kernel density estimation and recent advances. Biostatist. Epidemiol. **1**(1), 161–187 (2017)
6. Daganzo, C.F.: Traffic flow theory. In: Fundamentals of Transportation and Traffic Operations, pp. 66–160. Emerald Group Publishing Limited (1997). https://doi.org/10.1108/9780585475301-004. https://doi.org/10.1108/9780585475301-004
7. Dong, G., et al.: Graph neural networks in IoT: a survey. ACM Trans. Sen. Netw. **19**(2), 1–50 (2023)
8. Eslamlou, G.B., Jung, A., Goertz, N.: Smooth graph signal recovery via efficient laplacian solvers. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5915–5919 (2017)
9. Gerlough, D., Huber, M.: Traffic Flow Theory: A Monograph. National Research Council. Transportation Research Board, National Research Council (1975)
10. Jiang, W., Luo, J.: Graph neural network for traffic forecasting: a survey. Expert Syst. Appl. **207**, 117921 (2022)
11. Kondor, R.I., Lafferty, J.: Diffusion kernels on graphs and other discrete structures. In: Proceedings of the 19th International Conference on Machine Learning, vol. 2002, pp. 315–322 (2002)
12. Martínez Márquez, R.A., Patanè, G.: Graph node scoring for the analysis and visualisation of mobility networks and data. Urban Sci. **8**(4), 155 (2024)
13. Ni, D.: Traffic Flow Theory: Characteristics, Experimental Methods, and Numerical Techniques. Butterworth-Heinemann (2015)
14. Romero, D., Ma, M., Giannakis, G.B.: Kernel-based reconstruction of graph signals. IEEE Trans. Signal Process. **65**(3), 764–778 (2017)
15. Smola, A.J., Kondor, R.: Kernels and regularization on graphs. In: Learning Theory and Kernel Machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, 24-27 August 2003. Proceedings, pp. 144–158. Springer (2003)
16. Tabibiazar, A., Basir, O.: Kernel-based optimization for traffic density estimation in its. In: 2011 IEEE Vehicular Technology Conference (VTC Fall), pp. 1–5 (2011)
17. Wang, F., Li, T., Du, Y.: Kernel-based sampling of graph signals via graph sampling expansion theorem. In: 2022 7th International Conference on Signal and Image Processing (ICSIP), pp. 1–5 (2022)

18. van Wee, B., Annema, J., Banister, D.: The Transport System and Transport Policy: An Introduction. Edward Elgar (2013)
19. Xiao, J., Wei, C., Liu, Y.: Speed estimation of traffic flow using multiple kernel support vector regression. Phys. A **509**, 989–997 (2018)
20. Zhang, G., Wang, Y.: A gaussian kernel-based approach for modeling vehicle headway distributions. Transp. Sci. **48**(2), 206–216 (2014)

Resilience and Disruption in Mexico's Intermunicipal Mobility Network: A Component Decomposition Approach

Erika Cruz-Bonilla[1], Issa Moussa Diop[2], Christian Wolff[3], Hocine Cherifi[4], and Maribel Hernández-Rosales[1](✉)

[1] Center for Research and Advanced Studies of IPN, Irapuato Unit, Libramiento Norte Carretera Irapuato León Kilómetro 9.6, 36821 Irapuato, Guanajuato, Mexico
{erika.cruz,maribel.hr}@cinvestav.mx
[2] Université Cote d'Azur, DS4H, I3S Nice, France
issa-moussa.diop@univ-cotedazur.fr
[3] Mainz University of Applied Sciences, Lucy-Hillebrand-Straße 2, 55128 Mainz, Germany
Christian.Wolff@HS-Mainz.de
[4] Université De Bourgognem, ICB, UMR 6303 CNRS, Dijon, France
hocine.cherifi@u-bourgogne.fr

Abstract. Mexico's intermunicipal mobility network reveals a sharp contrast in the pandemic's effects. Using 731 daily networks from 2020 to 2021 and a component decomposition approach, the analysis shows that local components, representing short-distance travel, were heavily disrupted by mobility restrictions, leading to reduced connectivity and smaller community sizes. Meanwhile, global components, reflecting long-distance movement, remained resilient and structurally stable. These findings highlight how external shocks, like COVID-19 pandemic, proportionately impact local versus global mobility, revealing the robustness of long-distance networks in times of crisis.

Keywords: human mobility · mobile phone data · component structure · COVID-19

1 Introduction

Mobility networks are a powerful proxy for understanding the underlying structures and dynamics of societies. These networks, which encompass transportation systems, communication infrastructures, and migration patterns, reflect not only the physical movement of people and goods but also the social, economic, and cultural interactions that shape human behavior. Understanding how mobility patterns emerge, evolve, and impact the overall structure of a network is essential for optimizing efficiency, enhancing resilience, and reducing system-wide vulnerabilities. Data sources for mobility networks come from various sectors [9,16] such as mobile phone location data, GPS tracking in transportation

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 279–291, 2025.
https://doi.org/10.1007/978-3-031-82439-5_23

systems, social media and apps, OpenStreetMap, and city-specific data portals, and so on.

Mobility networks can be analyzed as complex networks at macroscopic, mesoscopic, and microscopic levels. Shi et al. [20] utilized mobile phone data to examine spatial distributions and human mobility patterns at the community level, identifying three spatial community patterns with distinct movement characteristics between urban areas and suburbs. Hossmann et al. [11] conducted a network-based analysis of mobile phone data, revealing that user mobility networks exhibit small-world characteristics and strong modularity, with consistent weight and degree distributions at both community and global levels. Additionally, Sandoval et al. [2] employed a network-based approach to analyze survey data on urban-rural issues in Bahía de Los Ángeles, Mexico, highlighting indirect influences among issues, parallels to recent real-life events, and notable behavioral differences between rural and urban contexts.

In [10], Fontanelli et al. presented a dataset of daily intermunicipal origin-destination networks in Mexico for 2020 and 2021. They analyzed these networks at macroscopic, mesoscopic, and microscopic levels, finding that mobility patterns were influenced by COVID-19 restrictions and population size. Key findings include a pre-pandemic mobility peak followed by a decline, a stable core community of municipalities, and a negative correlation between centrality variation and population size, with smaller municipalities showing more variability. Similarly, statistical analysis of the Federal Statistical Office of Germany [4] showed a decrease in mobility for the beginning of 2020 and 2021 whereas the long-distances routes were effected the most. Huang et al. [12] analyzed georeferenced tweets to see how global collaborative efforts in reducing human mobility are reflected from the user-generated information. The data was divided into single day and cross day trips. They found that the normalized mobility index showed the lowest values for single day and cross day trips at the end of march 2020. They also found country-level discrepancies in the time series of the two measures. Arambepola et al. [1] analyzed zip code-level mobility data from 26 US cities between February to August 2020 using Bayesian models to characterise the initial decrease in mobility and mobility patterns. They found that long-distance routes and higher-income subscribers, but not age, were associated with greater mobility reductions.

This paper analyzes Mexico's intermunicipal mobility network using the mesoscopic component structure [8], which divides the network into dense segments known as local components, representing short-distance movements, and global components that connect these segments, representing long-distance movements. This framework improves the visualization of mobility flows at both local and global scales. By applying this method to mobility data from before, during, and after the COVID-19 pandemic, the study aims to assess the impact of external factors like health restrictions and urbanization on mobility patterns, providing deeper insights into the dynamics of movement between dense areas and their connected regions.

2 Data and Methods

Network Source. Intermunicipal networks were obtained from [10], where networks were constructed using anonymized mobile phone location data of Mexico. Network files are available on https://osf.io/gwq6u/. Nodes are the municipalities or metropolitan zones in Mexico identified by the code assigned by the National Institute for Statistics, Geography and Informatics in Mexico (INEGI). Edges are observed mobility of devices from node i to node j. Meanwhile, weights are the normalized number of devices moving between nodes. Therefore, networks are weighted and directed networks.

Community Detection Algorithm. To extract the component structure, communities must first be identified within each network. Louvain community detection algorithm [3] was used on its implementation in networkx library in Python. Community detection was executed without network weights. The procedure involves two iterative steps. Initially, each node is treated as its own community. In the first step, the algorithm calculates the modularity gain that would result from moving each node into the community of one of its neighbors. This step is repeated until no further modularity improvements can be made. In the second step, a new network is constructed, where the identified communities from the first step are reduced to single nodes. The process continues until modularity can no longer be improved.

Local and Global Components Extraction. The extraction of global and local components follows the methodology outlined in [5,8]. This process is based on the premise that networks contain dense, localized areas where nodes primarily interact within their community. The network is divided into two sets: local components, which are dense areas sharing information internally, and global components, which consist of nodes and links connecting different local components. The extraction algorithm involves three main steps:

1. Identify Dense Regions: Use a community detection algorithm to reveal the community structure within the network.
2. Extract Local Components: Isolate local components by removing inter-community links from the identified community structure.
3. Extract Global Components: Remove intra-community links and any isolated nodes to uncover the global components.

K-Core. The k-core [14] of a network is a sub-network where each node has at least k connections. To extract the maximum global k-core, nodes were iteratively removed from the biggest global component subnetwork for each day. The process began with the 1-core, which included the entire network since every node had at least one link. Next, the 2-core was formed by removing all nodes with only one connection. The 3-core was then obtained by eliminating nodes with two links in the 2-core, and this process continued until reaching the maximum k-core, beyond which it was impossible to form a (k+1)-core.

Network Topological Properties. Network properties were calculated using Python package igraph [6], where networks were uploaded as directed and weighted network objects. Network properties were calculated for whole networks and all local and global subnetworks of the corresponding date. Calculated properties for the networks were: density, unweighted diameter, mean of all nodes in-out strength (total strength), average unweighted shortest path, undirected transitivity and assortativity by degree.

Map Generation. Shapefiles with geographical delimitation of the municipalities, states and country were obtained from the INEGI, from the Geostatistical Framework for the Population and Housing Census, 2020 edition [13]. Shapefiles for the metropolitan zones delimitation were obtained from [7]. Maps were generated using the sf package in R [17].

3 Results

3.1 Topological Characteristics of the Intermunicipal Networks of Mexico

Our source network set consisted of 731 networks detailing the human mobility between municipalities in Mexico in the time period between 2020 and 2021, one network per day [10]. Network nodes consisted of the 2,446 municipalities and 107 metropolitan zones.

As a first look into the intermunicipal networks, Fig. 1 presents the distribution of six topological properties of the networks over time. Each subplot focuses on a different network property: density, diameter, total strength, average shortest path, transitivity, and assortativity. The evolution of these topological properties, seems to be divided into three distinct periods of significant change. The first period spans from early January 2020 to the end of July 2020. The second period extends from August 2020 to April 2021. Finally, the third period begins in April 2021 and continues through the end of 2021.

During the first period, the networks' density and total strength is relatively high, especially in February 2020. Meanwhile, diameter and average shortest path length take relatively low values when compared to the second period. Then, on the second period, density and strength show a slight decrease, while diameter and average shortest path increase, reaching its peak around mid-2020. In this period, longer paths are required to connect different parts of the network and suggesting maximum dispersion in the network. Towards the last period, there is a slight increase in density and a steep increase in total strength. Additionally, the diameter and shortest path length return to their initial state before reaching their smallest value towards the end of 2021. Assortativity remains negative for all time periods, indicating that in general nodes had a tendency to connect with dissimilar nodes, but it increases notoriously near zero in the second period, suggesting a slight shift towards connection between similar nodes.

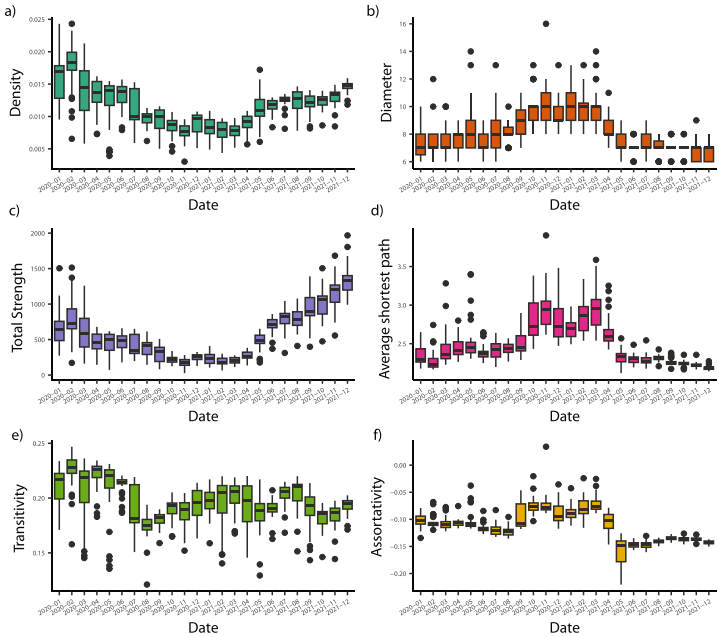


Fig. 1. Distribution of network properties for complete intermunicipal travel networks, each box shows the statistics grouped by month. a) Density, b) Diameter (unweighted), c) Total strength, d) Average shortest path without weights, e) Undirected transitivity and f) Assortativity by degree.

Altogether, these changes suggest a decrease in traffic during the second period compared to the first, as evidenced by longer distances and fewer connections in the networks of this time frame. This period also includes the months with the lowest traffic. In contrast, the third period, which corresponds to the months with the highest traffic, shows a return to more compact networks, indicating that travel became easier during this time.

Despite variations in other properties, transitivity remains consistently low and stable, suggesting that the local structure of the network has remained largely unchanged throughout the different periods.

3.2 Component Decomposition of Intermunicipal Networks

To analyze the underlying communities in Mexico defined by population transit and to better understand how they interact at the intra and inter regional level, we set out to identify the local and global components of the networks. In our context, local components denote short displacements within communities and global components reflect long displacements between them. Each component is a subnetwork and we can explore their topological properties (macro-scale), k-core (meso-scale), and the influence of the nodes (micro-scale). Here, we first

analyze the local components and global components separately, afterwards, we analyze the core of the global components.

Local Components. Given the fluctuation of transit across days, different numbers of local components were detected when comparing all networks (Fig. 2). The total number of local components varied between 7-40, with the average number of local components being 13. The maximum number of local components was observed on November 30th, 2020, meanwhile the minimum number of local components was registered on May 26th, 2020.

Sizes of the local components, that is how many municipalities are included within each community, vary greatly when considering all components identified across all networks. Most components identified are of small size (< 100 municipalities). 52% of the components belong to this size range. The next most common size for components is on the range of 200-300 municipalities, with 23% of the components being of this size. The maximum range for the component size identified is around 700-600 municipalities. This component size may seem extremely high given that most states in Mexico have on average 77 municipalities. However, municipality density in Mexico is also quite diverse. For example, five states concentrate more than half of the municipalities in all Mexico: Oaxaca (570), Puebla (217), Veracruz (212), Jalisco (125) and State of Mexico (125) [13]. The most municipality dense state, Oaxaca, actually appears consistently as a local component (Fig. 2) and might be the main reason of the existence of the big local components.

Additionally, the local components size was observed to remain stable through time. In each day/network the trend continues where most local components identified are within the small and mid-size range. Deviations from these observations seem to be only present during a drop in June 2020, and with peaks from December 2020 to February 2021, and May to July 2021, where the mean local component size decreases and increases, respectively. Both changes in component size might indicate sudden drops or increases in mobility between localities, where these changes match events during the pandemic, such as the start of the government call for quarantine (June 2020) [19], the Christmas holidays, and the relaxation of the transit restrictions given the low contagion incidence during May-June 2021 [18].

Municipality composition of the local components also remains somewhat stable across dates (Fig. 3). The variation in component size and composition seems to be driven mainly by network size and the appearance of subcommunities within bigger components (Figs. 2,3). For example, the small components that arise within the local components that corresponds to the state of Oaxaca, and along the central region of the country, on November 17th. In the first scenario, intermunicipal network size changes through dates as some municipalities were not included as nodes within networks whenever the travel weight did not exceed a certain threshold [10]. In days with low mobility, this causes the network to be sparse, with isolated nodes, and therefore a high number of components of small size (Fig. 2, Ex. 2020-11-30). For the later scenario, sub communities within

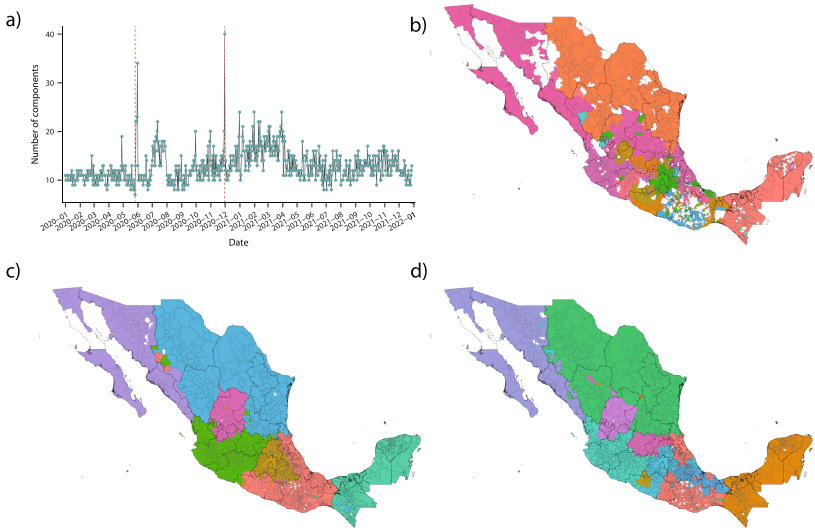


Fig. 2. a) Variation of the number of local components identified across time. Dot lines show the dates for which the maximum (2020-11-30) and minimum (2020-05-26) number of components were identified. b) Map of the local components identified for 2020-11-30. c) Map of the local components identified for 2020-05-26. d) Map of the local components identified for 2020-11-17, random day selected with average number of components (n=13)

previous local components are made evident in days with high traffic and mostly around metropolitan zones.

Global Components. The study of the global components in Mexico is also promising. These components represent the municipalities that serve as interconnections between the identified communities. In Mexico, the number of global components identified across all networks ranged from 1-19 global components by day, with a mean number of global components being four. The highest number of global components identified was 19, on November 30th 2020. This is coincident on the day with the highest number of local components (Fig. 2), further solidifying that this date corresponds to a date with fragmented and localized mobility. When analyzing the distribution of municipalities across global components, it is noticeable that despite having a variable number of global components through time, most municipalities belong to a principal large global component. This trend continues even on days when mobility is low and few municipalities are within the network. The global components that arise apart from the biggest global component belong to isolated connections between pairs of communities. This indicates that interaction between most communities can be summarized by the analysis of the biggest global component identified. The Fig. 4 consists of six box plots representing the evolution of various network properties such

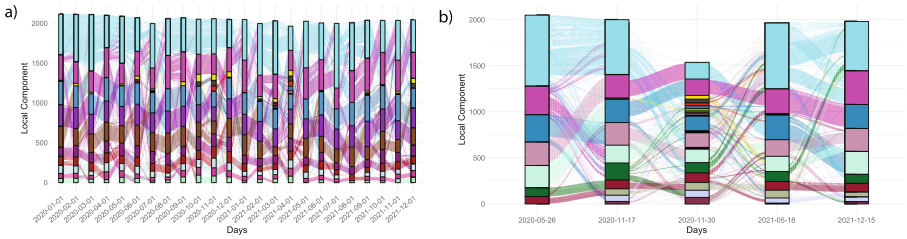


Fig. 3. Alluvial plots for a) the local components identified at the beginning of each month during all years examined, and b) the local components identified from networks at selected dates. For both plots, each thread represents a municipality or a metropolitan zone, colored according to the local component they belong to. For each day, nodes are grouped according to their local component.

as density, diameter, component size, average shortest path, transitivity, and assortativity over time for the largest global component. Note that the periods we identified on the whole networks topological analysis are not as visible in the context of the global components. Only the sizes of the largest global component, and assortativity follow this trend. The density tends to fluctuate but appears to decline slightly over time. By 2021, density becomes more stable, with lower values overall. The largest global component is two times less dense than the entire network over all dates sampled. The diameter values are mostly stable throughout the observation period, with only a slight perturbation during June-July 2020. Nonetheless, the diameter of the largest global components is on average bigger than that of the whole networks. The average shortest path also remains relatively stable over time, hovering around 2 to 3. These values are comparable to those of the first and last periods of the whole networks. Transitivity shows a clear decline over time, starting from 0.12 to 0.07. The decrease suggests that the largest global component becomes less clustered, with fewer triadic closures. Generally, long-distance journeys by road are direct. The assortativity values show that, over time the largest global component is disassortative like the entire network. However, the global components tend to be more disassortative than the whole network. Indeed, the assortativity values of the largest global component are comparable to the values of the third period of the network, which is most disassortative.

Core Analysis of Global Components. Within the global component networks, there are nodes that are central in the connectivity due to their higher degree and might indicate municipalities that are key in the traffic within and between communities. These nodes can be identified by k-core analysis of the daily global component networks. A depiction of the localization of the k-core municipalities in selected dates is shown in Fig. 5. This selection of dates, where the number of local and global components reach a minimum and maximum, shows that k-core global components are highly variable.

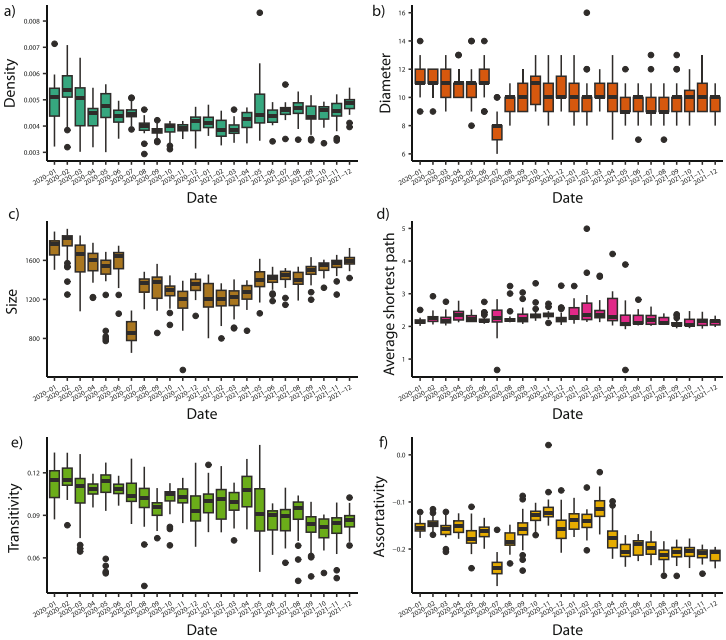


Fig. 4. Distribution of network properties of the biggest global component network across days, each box shows the statistics grouped by month. a) Density, b) Diameter (unweighted), c) Largest global component size, d) Average shortest path without weights, e) Undirected transitivity and f) Assortativity by degree.

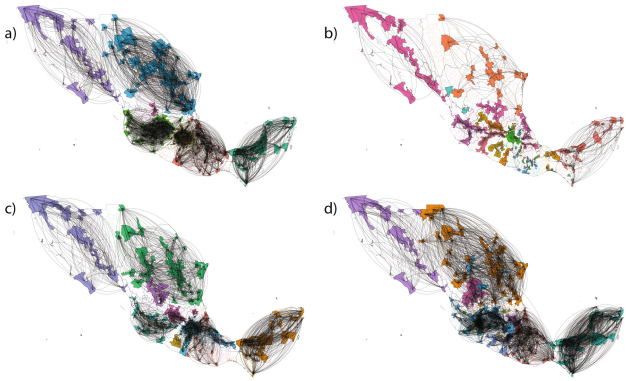


Fig. 5. Map of the k-core global components identified for selected days: a) 2020-05-26, b) 2020-11-30, c) 2020-11-17, and d) 2021-05-18 For each map, the k-core global components are solid colored, according to the local component they belong to. Edges between global components are shown without weight.

Given the vastness of networks we have analyzed, it is interesting to check if, despite the variability, there are municipalities that are repeatedly classified as core global components. As expected, most of the core global components are selected as such in only a handful of days (less than 10 days, Fig. 6). Nonetheless, there is a set of nodes that appears as a core global component in more than 700 days. This group of nodes are practically all metropolitan zones. The only exception being Irapuato municipality. In fact, 30 of the 107 metropolitan zones are consistent k-core global components.

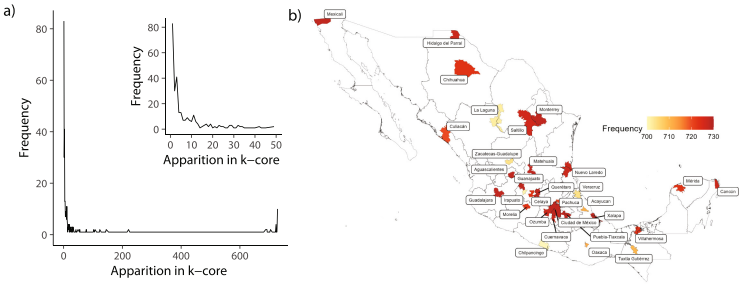


Fig. 6. a) Frequency of the number of times a given municipality appears as a k-core global component across all networks. A zoom in is shown for frequency < 50 days (Inset). b) Map showing the municipalities with higher apparition rates as k-core elements of the global components (> 700 days).

4 Discussion and Conclusion

The analysis of Mexico's intermunicipal mobility networks throughout 2020 and 2021 revealed three distinct patterns corresponding to the pre-pandemic, pandemic, and post-pandemic phases. These shifts in mobility are closely linked to the behavioral changes driven by the COVID-19 pandemic and government-imposed restrictions. By decomposing the network into local and global components, we captured the differential impact of these phases on both short- and long-distance travel, offering a nuanced understanding of mobility patterns under external stress.

The pre-pandemic period, marked by normal activity and routine mobility, showcased dense, well-connected networks. Once the pandemic struck and restrictions were imposed (March 2020) [19,21], a marked reduction in mobility emerged, particularly within local components, which represent short-distance travel. This period, referred to as the "pandemic mobility status," is characterized by a sharp decline in density and a spike in both the network's diameter and average shortest path length, reflecting greater distances between connected communities and a more dispersed network structure. Local networks were significantly fragmented during this time, with fewer connections and a lower number

of communities as mobility restrictions took hold. These changes illustrate the vulnerabilities of local mobility networks to sudden external shocks, where even minor disruptions ripple through smaller, localized systems.

Interestingly, the global components—representing long-distance connections between municipalities—proved far more resilient. Despite a reduction in overall traffic, the long-distance connections largely maintained their structural integrity throughout the pandemic. This resilience is a key finding, underscoring the robustness of critical transport and mobility corridors that are less affected by short-term restrictions. Metropolitan zones, in particular, played a pivotal role in maintaining these global networks, serving as the backbone of long-distance travel throughout the studied period. This stability highlights the reliance on these hubs for maintaining essential connectivity, even during times of crisis.

The post-pandemic period brought a gradual return to pre-pandemic mobility patterns. With vaccination campaigns [2] and the easing of restrictions, mobility bounced back, and networks became more cohesive once again. The resurgence of local components, reflected by increased density and shorter average paths, signified a reactivation of short-distance travel, particularly in smaller municipalities. For example, on May 26th 2020, the day with least local components, there is evidence to a partial return to work-related activities which could explain the high mobility and connectivity observed during this time period [18]. Global components, meanwhile, exhibited little change, reinforcing their role as stable structures within Mexico's mobility network.

The geographic analysis of local components further enriched our understanding of regional mobility patterns. Mexico's local mobility consistently clustered into approximately 8-10 zones, roughly corresponding to well-known historical, cultural, and geographical divisions. These zones include the Northeast (Baja California, Baja California Sur, Sinaloa, Sonora), Northwest (Durango, Chihuahua, Coahuila, Nuevo León, Tamaulipas, San Luis Potosi), Central Lowlands (Guanajuato, Queretaro, Aguascalientes, Zacatecas), Central Valley (Mexico City, State of Mexico, Tlaxcala, Puebla, Hidalgo, Morelos), West (Nayarit, Jalisco, Colima, Michoacan), East (Guerrero, Oaxaca, Veracruz), and Southeast (Chiapas, Tabasco, Campeche, Quintana Roo, Yucatan) regions. This observation suggests that local mobility patterns, despite disruptions, remain closely tied to underlying socio-cultural and geographic factors [15,22].

Overall, the component decomposition analysis provides a unique lens through which to understand how shocks like the COVID-19 pandemic affect mobility at different levels. Local networks were far more susceptible to disruption, while global networks —anchored by metropolitan zones—demonstrated remarkable resilience. This duality offers insights for future urban planning and policy, emphasizing the need for strengthening local mobility systems to reduce vulnerability, while maintaining robust long-distance connections to ensure continuity during crises.

Acknowledgements. This work was supported by the Fondo Conjunto de Cooperación México-Uruguay from the Agencia Uruguaya de Cooperación Internacional and the Agencia Mexicana de Cooperación Internacional para el Desarrollo.

Code Availability. The code used for the analysis and extraction of local and global components from the intermunicipal mobility network of Mexico is available upon request. The associated scripts are provided in a repository hosted on GitHub, which can be accessed at the following link: https://github.com/biocomplexnet/Mexico-Intermunicipal-Network-Decomposition

References

1. Arambepola, R., Schaber, K.L., Schluth, C., Huang, A.T., Labrique, A.B., Mehta, S.H., Solomon, S.S., Cummings, D.A.T., Wesolowski, A.: Fine scale human mobility changes within 26 US cities in 2020 in response to the COVID-19 pandemic were associated with distance and income. PLOS Global Public Health **3**(7), e0002151 (2023). https://doi.org/10.1371/journal.pgph.0002151. https://journals.plos.org/globalpublichealth/article?id=10.1371/journal.pgph.0002151. Publisher: Public Library of Science
2. Bello-Chavolla, O.Y., et al.: Effectiveness of a nationwide Covid-19 vaccination program in mexico against symptomatic Covid-19, hospitalizations, and death: a retrospective analysis of national surveillance data. Int. J. Infectious Dis. **129**, 188 (2023). https://doi.org/10.1016/J.IJID.2023.01.040. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9918316/
3. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. **2008**(10), P10008 (2008)
4. Bundesamt, S.: Mobilitätsindikatoren auf Basis von Mobilfunkdaten (2024). https://www.destatis.de/DE/Service/EXSTAT/Datensaetze/mobilitaetsindikatoren-mobilfunkdaten.html
5. Cherifi, H., Palla, G., Szymanski, B.K., Lu, X.: On community structure in complex networks: challenges and opportunities. Appl. Netw. Sci. **4**, 117 (2019). https://doi.org/10.1007/s41109-019-0238-9. https://doi.org/10.1007/s41109-019-0238-9
6. Csárdi, G., Nepusz, T.: The igraph software package for complex network research (2006). https://api.semanticscholar.org/CorpusID:16923281
7. de Desarrollo Social, S., de Estadística y Geografía, I.N., de Poblacion, C.N.: Delimitación de las zonas metropolitanas de México (2010). http://www.conapo.gob.mx/es/CONAPO/Datos_Abiertos_Delimitacion_de_Zonas_Metropolitanas
8. Diop, I.M., Cherifi, C., Diallo, C., Cherifi, H.: Revealing the component structure of the world air transportation network. Appl. Netw. Sci. **6**, 1–50 (2021). https://doi.org/10.1007/S41109-021-00430-2/TABLES/19. https://link.springer.com/articles/10.1007/s41109-021-00430-2https://link.springer.com/article/10.1007/s41109-021-00430-2
9. Flores-Garrido, M., et al.: Mobility networks in greater Mexico city. Sci. Data **11**(1), 84 (2024)
10. Fontanelli, O., et al.: Intermunicipal travel networks of Mexico during the Covid-19 pandemic. Sci. Rep. **13**, 1–12 (2023). https://doi.org/10.1038/s41598-023-35542-5. https://www.nature.com/articles/s41598-023-35542-5
11. Hossmann, T., Spyropoulos, T., Legendre, F.: A complex network analysis of human mobility, pp. 876–881 (2011). https://doi.org/10.1109/INFCOMW.2011.5928936

12. Huang, X., Li, Z., Jiang, Y., Li, X., Porter, D.: Twitter reveals human mobility dynamics during the COVID-19 pandemic. PLOS ONE **15**(11), e0241,957 (2020). https://doi.org/10.1371/journal.pone.0241957. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0241957. Publisher: Public Library of Science
13. INEGI (Instituto Nacional de Estadística, Geografía e Informática): Marco Geoestadístico. Censo de Población y Vivienda 2020. Tech. rep. (2020)
14. Kong, Y.X., Shi, G.Y., Wu, R.J., Zhang, Y.C.: k-core: theories and applications. Phys. Rep. **832**, 1–32 (2019)
15. Martínez, B.G.: Las regiones de México: breviario geográfico e histórico, vol. 5. El Colegio de Mexico AC (2008)
16. Pappalardo, L., Manley, E., Sekara, V., Alessandretti, L.: Future directions in human mobility science. Nature Comput. Sci. **3**(7), 588–600 (2023)
17. Pebesma, E.: Simple Features for R: Standardized support for spatial vector data. R Journal **10**(1), 439–446 (2018). https://doi.org/10.32614/RJ-2018-009
18. SEGOB: Diario Oficial de la Federación (DOF): 29/05/2020. Official Journal of the Federation of the government of Mexico (2020). https://www.dof.gob.mx/nota_detalle.php?codigo=5594138&fecha=29/05/2020
19. SEGOB: Diario Oficial de la Federación (DOF): 30/03/2020. Official Journal of the Federation of the government of Mexico (2020). https://www.dof.gob.mx/nota_detalle.php?codigo=5590745&fecha=30/03/2020
20. Shi, L., Chi, G., Liu, X., Liu, Y.: Human mobility patterns in different communities: a mobile phone data-based social network approach. Ann. GIS **21**, 15–26 (2015). https://doi.org/10.1080/19475683.2014.992372
21. Suárez, V., Quezada, M.S., Ruiz, S.O., Jesús, E.R.D.: Epidemiology of Covid-19 in mexico: From the 27th of february to the 30th of april 2020. Revista Clinica Espanola **220**, 463 (2020). https://doi.org/10.1016/J.RCENG.2020.05.008. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7343663/
22. Zepeda, R.V., de México. Instituto de Geografía, U.N.A.: Las regiones climáticas de México I.2.2. Instituto de Geografía, UNAM (2005). https://books.google.com.mx/books?id=qwLAtAEACAAJ

The Impact of Economic Factors on Brain Circulation Patterns

Afra Temiz, Mehmet Eren Bulut(✉), and Mehmet Burak Akgun

TOBB University of Economics and Technology, Ankara, Turkey
m.akgun@etu.edu.tr

Abstract. The concept of brain drain has long been a significant topic of discussion. Advancements in technology have significantly facilitated transportation and communication which in turn has led to increased human mobility over the last few decades. In particular, traveling to different countries for education and employment has become increasingly widespread. This study extracts academic talent migration data from a major academic publishing dataset and examines brain drain trends between countries in relation to their economic conditions...

Keywords: brain circulation networks · migration patterns

1 Introduction

Brain drain has emerged as a pivotal concern, transcending geographical borders and impacting nations at various stages of development. The migration of highly skilled workers, particularly within the academic community, has profound implications for both the countries losing talent and those gaining it.

In a world characterized by heightened connectivity and mobility, researchers have long surpassed the limitations of geographical boundaries. The academic diaspora constantly shifts, with minds traversing continents and institutions pursuing diverse opportunities and better resources.

Researchers are inherently attracted to more prosperous nations because of appealing economic prospects, advanced research infrastructure, and the opportunity to contribute to innovative research. Unfortunately, this migration amplifies existing inequalities, contributing to a global scenario where the advantages of highly skilled labor are unevenly distributed among a limited number of countries that engage in research to further their specific needs.

The gravitational pull of developed nations can be seen as a form of economic and intellectual exploitation. Highly skilled workers from less fortunate countries often find themselves contributing to the prosperity and advancement of countries that are already advanced and prosperous.

In the scope of this research, we attempt to unravel the dynamics of researchers' brain drain by tracing their movements from publicly available profile data. We aim to identify the contributors to brain drain and the beneficiaries.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 292–303, 2025.
https://doi.org/10.1007/978-3-031-82439-5_24

2 Methodology

Researchers' profiles are scrapped from the site "ORCID" where a unique and persistent ID is assigned to individuals. ORCID is an international, interdisciplinary, open, non-proprietary, and not-for-profit organization created by the research community. ORCID provides a persistent digital identifier (an ORCID iD) that distinguishes individuals from other researchers and a record that supports automatic links among all professional activities.

2.1 Data Collection

The data is stored in compressed form in documents encoded with the ORCID iD code created for each researcher. For each researcher, there is a summary XML file containing the following information:

- Start
- ISO 3166-1 alpha-2 country code
- Institution
- Role Title

Since XML files contain more information that will not be used in this study, data cleaning was performed. All researchers' data were saved in CSV files in the following format:

ORCID iD||Identifier||Year||Country Code||Institution Name||Role Title

Identifier: "edu" for education, "emp" for employment

2.2 Data Properties

Data were gathered from 4,663,272 distinct researchers in total. Of these researchers, 3,810,541 were located in a single country. This means, that when doing a brain drain study, the movements of the remaining 852,731 researchers will be the major focus.

There are 250 countries represented among these researchers. These countries are all 249 officially assigned ISO 3166-1 alpha-2 country codes and Kosovo. This scope shows that the data obtained is global. Which makes thorough analysis across various geographic and political entities possible.

There are 13124578 entries, 6310309 related to employment, and 6814269 related to education. This large dataset provides a substantial amount of information to analyze trends in academic and professional movements.

While institution names and role titles were collected as part of the dataset, the lack of consistency in naming conventions made it challenging to analyze this data effectively. As a result, this component of the dataset was excluded from the analysis in this paper.

3 Results and Analysis

When conducting the analysis, general patterns were first examined. Knowing the main lines of the data is important in terms of predicting what to do in the next stages. To make these predictions more consistent, an analysis perspective that goes from general to specific was followed. This approach ensures a comprehensive understanding of the overall trends before delving into more nuanced aspects of the data.

In the first stage of the analysis, general lines will be found, then the most effective countries in brain drain will be determined. Analyses will continue for these countries, and detailed inferences will be made by considering the economic and social effects. By focusing on these key countries, the analysis can uncover the drivers behind significant migration patterns and provide insights into policy implications that could address brain drain issues.

3.1 General Data Insights

Distribution of Inputs by Year. The first general aspect to be examined in the analysis is the distribution of inputs by year. This information tells about in which years there is more mobility in education and work life. These values cover all inputs. In other words, they do not provide definitive information about whether there is a brain drain. Because a researcher may provide a new input to his or her own country. The graph of the distribution of inputs by year is given below. (see Fig. 1)

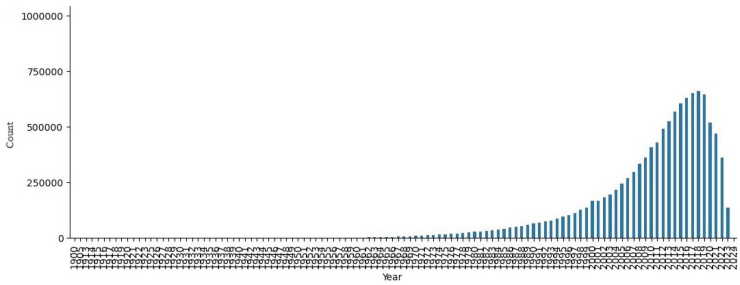


Fig. 1. Distribution of Inputs by Year

As seen in Fig. 1, the first entry is from 1900, while the last entry is from 2024. Although there has been a steady increase in entries since the late 1970s, it can be said that the real increase started in 2000. The highest entry was seen in 2018.

This increase likely reflects the globalization of educational and employment opportunities and advances in technology and communications.

Distribution of Inputs by Countries. Another aspect to look at is the distribution of countries in the inputs. This makes it easier to analyze the countries where researchers are most present and visited. As in the previous analysis, although this does not provide direct information about brain drain, it is an important basic analysis to see the general aspects of the countries.

By identifying the most frequently occurring countries, we can prioritize which regions to focus on for more detailed analysis. This foundational step also helps in understanding the global landscape of researcher mobility.

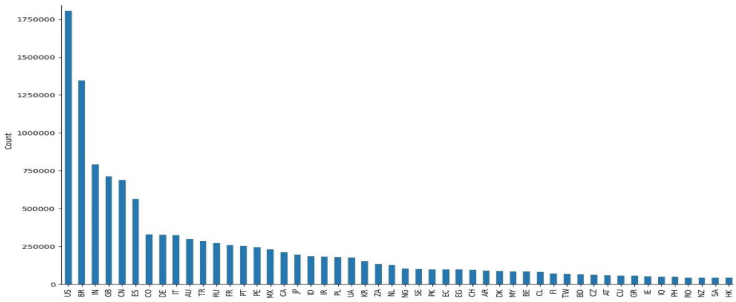


Fig. 2. Distribution of Inputs by Country

In Fig. 2 top 50 countries are shown for better visibility. The countries that have the most entries are the United States, Brazil, India, the United Kingdom, and China. It is likely that these countries are important centers for brain drain.

More analysis is needed to gain a better understanding of the countries' roles in brain drain. Thus, countries' degree centrality and betweenness centrality are examined, for this networkx library is used. A node is created for each country. Then for each ORCID iD, a set of countries that belong to that ORCID iD is written. If there is a move for that researcher, then an edge from source country to destination country is created.

For degree centrality, the degree (number of connections) for each node (country) in the graph is computed. This provides information about the countries with the highest brain drain and those that contribute the most to brain drain. The countries with the top 5 degree centrality values are:

1. United States
2. United Kingdom
3. Spain
4. Canada
5. Australia

When compared with Fig. 2, it is seen that some countries' degree centrality rank is lower. This generally means they mostly appeared in internal activities. They may not be a critical point of transit or interaction in the network.

By combining total appearances and degree centrality, it can be said that the United States, the United Kingdom, and Spain are the most important countries in the brain drain world.

For betweenness centrality how often a node appears on the shortest path between other nodes is computed. This helps to identify countries that serve as critical transit points between other countries.

1. United States
2. United Kingdom
3. Australia
4. Spain
5. Canada

The betweenness centrality is quite similar to degree centrality. The countries that appear in both have a key role in directly connecting other countries or acting as a bridge. These countries are central players in global connectivity.

Finding the Balance Between Countries. After finding the most important countries in the previous step, in this step total number of movements between country groups will be researched. It has been seen in the analyses conducted so far that some countries contribute more to brain drain. From now on, the point where the balance of brain drain movements is achieved will be tried to be found. To do this, the countries ranked according to their influence and separated into two groups.

To rank countries according to their influence, three things were considered: the number of entries for that country, the betweenness centrality value, and the degree centrality value. After the ranking, the countries at the top of the ranking were made into a group, and the other countries were made into a group. The number of movements between these two groups was examined until a balance was found. (see Fig. 3)

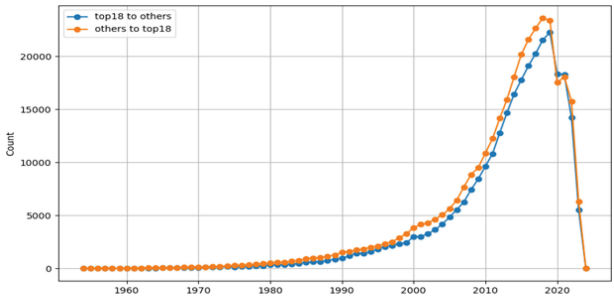


Fig. 3. Movement Comparison between top18 vs other countries

It has been seen that balance has been achieved between the 18 countries with the highest influence and the other countries. Figure 3 shows the comparison of

the brain drain movement from the top 18 countries to other countries and the brain drain from other countries to the top 18 countries over the years.

As can be seen in Fig. 3, the top 18 countries are in a significant position in terms of brain drain, at least as much as all other countries combined.

The following migration patterns are observed:

- Movements between the top 18 countries account for approximately 42% of all migration.
- Movements from the top 18 countries to other countries make up around 22% of all migration.
- Movements from other countries to the top 18 countries constitute about 24% of all migration.
- Movements between countries outside the top 18 represent approximately 12% of all migration.

This means that almost 90% of brain drain movements are related to the top 18 countries. Therefore, the rest of the analysis will focus on each of these 18 countries in detail and will focus on correlating brain drain patterns with the economic and social situations of these countries.

3.2 Detailed Analysis

The top 18 countries in the most influential countries list created in the previous step will be examined. World Bank's World Development Indicators database from 2014 to 2023 will be used for the analysis. Connections will be established between the GDP, inflation, and migration values obtained from the World Bank database and the number of incoming and outgoing researchers obtained from the analysis of ORCID data. The examination will be conducted by categorizing the countries into groups based on their 2023 GDP values. The first group will consist of the seven countries with GDP values of 3 trillion USD or higher. The second group will include six countries with GDP values between 1.5 and 3 trillion USD. All remaining countries will be placed in the third group.

First Group. This group includes the United States, China, Germany, Japan, India, the United Kingdom, and France. The ranking is given from the highest to the lowest GDP value.

The ranking of the countries according to the most influential countries list is as follows: United States (1), United Kingdom (2), India (4), China (8), France (9), Germany (10), Japan (16). The average ranking of the countries in this group is 7.14.

Patterns: Percentages, Factors and Comparisons. In order to examine these countries, firstly the movement patterns will be discussed. For this purpose, whether the researchers stayed in their own countries was checked. To make this analysis, the country of the researchers' first entries was assumed to be their own country. Then, it was examined whether the researcher had entries from

other countries other than his own country. As the last stage, the country of the researcher's last entry has been obtained.

Some researchers have never left their home country, while others have gone to other countries. Among the researchers who went, there are those who returned to their home country and those who did not. Table 1. shows the proportion of researchers in this group who stayed, went and returned, and those who did not return.

Table 1. Researcher Mobility Percentages by Country

Country	Stayed(%)	Returned (%)	Did Not Return (%)
United States	87.85	3.07	9.08
China	82.06	5.44	12.50
Germany	73.71	7.93	18.36
Japan	83.87	6.23	9.90
India	83.35	3.17	13.48
United Kingdom	72.17	7.14	20.69
France	69.83	9.19	20.98

As can be seen in Table 1, all countries except one (Germany) follow a pattern. The higher the GDP of a country, the higher the number of researchers staying in the country. India is the country that has the lowest ratio of returned to did not return. At the same time, India is the country with the highest inflation rates in this group. These show how important economic conditions are in brain drain, to gain more insight a little more country-based detail will be examined.

The first finding that needs to be mentioned is the similarity between the United States and the United Kingdom. These two countries are the primary recipients of brain drain. They are the only countries where the ratio of outgoing researchers/incoming researchers changes in proportion to inflation through the years.

Another significant observation is the ratio of the number of incoming researchers to the country's net migration value. While this ratio is around 1–5% in all other sixteen countries, it is around 20% for Germany and France. In other words, a significant part of people migrating to these countries are researchers. This can be related to them being the biggest economies in the European Union.

One of the other findings is the COVID's similar impact on China and Japan. Previously, the ratio of incoming researchers to outgoing researchers was balanced for Japan and increasing for China. After COVID, this ratio dropped significantly for both, unlike other countries. And the ratios are yet to reach their previous high levels.

It is foreseeable that some countries receive more brain drain than others. The countries to which researchers in each country go for brain drain vary. In

this section, we will examine the countries to which the countries on the list send the most brain drain.

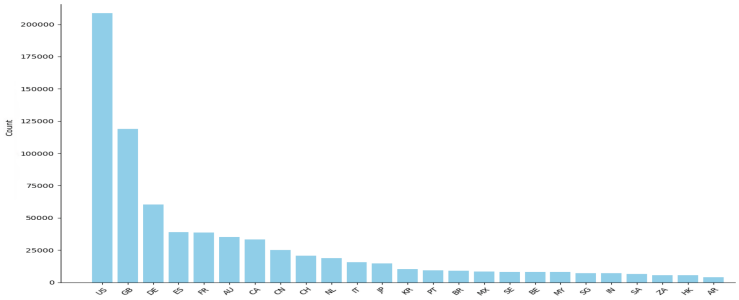


Fig. 4. Countries that receive most brain drain

Figure 4 shows countries that receive the most brain drain. The brain drain locations of the countries in this group will be examined and their compatibility with Fig. 4 will be analyzed.

Destination Trends in Brain Drain. Japan's most significant brain drain has been to the United States and China. For Germany, France, India, and China most significant brain drain has been to the United States and the United Kingdom. While Canada and China have been the primary destinations for the United States's brain drain. The United Kingdom's most notable brain drain has been directed toward Canada and Australia.

As supported by Fig. 4 United States and the United Kingdom are seen as popular brain drain countries. In addition, it has been seen in the case of Japan and the United States that geographical location affects the preferred countries for brain drain.

Second Group. The countries in this group are Italy, Brazil, Canada, the Russian Federation, Australia, and Spain. The ranking is given from the highest to the lowest GDP value.

The ranking of the countries according to the most influential countries list is as follows: Spain (3), Australia (5), Brazil (6), Canada (7), Italy (11), Russian Federation (15). The average ranking of the countries in this group is 7.83.

Patterns: Percentages, Factors and Comparisons. For this group, the percentages of those who stayed in their own countries, those who went and returned, and those who did not return will be examined too. Comparisons will be made with the first group. Thus, it will be possible to see how the GDP value affects these rates comparatively and more clearly. This will help to understand the impact of economic changes on movement trends more thoroughly.

When Table 1 and Table 2 are compared it is seen that although the GDP values of second group countries are lower, the average of stayed percentage is

Table 2. Researcher Mobility Percentages by Country

Country	Stayed(%)	Returned (%)	Did Not Return (%)
Italy	74.69	7.66	17.65
Brazil	91.46	3.24	5.30
Canada	71.62	6.67	21.71
Russian Federation	92.15	1.06	6.79
Australia	78.16	8.91	12.93
Spain	80.21	8.86	10.93

higher. This is the exact opposite of the pattern seen in the first group, it can be interpreted that in countries with lower GDP value, more people stay in the country because it is more difficult to go abroad for economic reasons.
Another detail supporting this finding is that Spain and the Russian Federation, which have the highest rate of researchers staying, are the countries with the highest inflation values in this group.

If we examine the countries in this group in a more country-by-country manner, the first finding is that the brain drain values of Canada and Australia have found a general balance. The brain drain in these two countries is not affected by economic factors and there is a balanced increase in the number of researchers coming to the countries. In addition, these two countries are the only two countries in this group where the number of incoming researchers is higher than the number of outgoing researchers through the years.

Brazil and the Russian Federation are the only countries that through the years incoming researchers have been always significantly lesser than outgoing researchers as stated before these two countries have the highest inflation rates which can be associated with low incoming researchers. One thing to add for the Russian Federation is among the countries examined so far, it has the lowest number of outgoing and incoming researchers. The number has never exceeded 1,000 for either category in any given year.

The last finding is the ratio of incoming researchers to outgoing researchers has been increasing steadily over the years in Italy and Spain. They are the only two countries in this group that have seen this ratio increase steadily each year.

Destination Trends in Brain Drain. Countries to which the countries in the second group send the most brain drain are as follows. For Spain, Australia, Canada, and Italy most significant brain drain has been to the United States and the United Kingdom. The main countries to which Brazil's brain drain has been directed are the United States and Portugal. The United States and Germany have been the primary destinations for the Russian Federation's brain drain.

The United States and the United Kingdom are considered prominent brain drain countries, as evidenced by Fig. 4, and supported by both the first and second groups. Moreover, the theory that chosen nations for brain drain are

influenced by physical location is supported by the case of Canada. Additionally, one thing that should not be overlooked in Brazil's case is the similarity in language.

Third Group. This group consists of the Netherlands, Turkiye, Colombia, Portugal, and New Zealand. The ranking is given from the highest to the lowest GDP value.

The ranking of the countries according to the most influential countries list is as follows: Colombia (12), Netherlands (13), Portugal (14), Turkiye (17), New Zealand (18). The average ranking of the countries in this group is 14.8.

Patterns: Percentages, Factors and Comparisons. Similar to the process in the first two groups, the rates of researchers who stayed in their country, went abroad and returned, and went abroad and did not return will be examined. Although Table 3 is not a striking detail when compared to the tables of the previous groups, one thing that can be added is that this group has the highest average rate of researchers who did not return to their country.

Table 3. Researcher Mobility Percentages by Country

Country	Stayed(%)	Returned (%)	Did Not Return (%)
Netherlands	71.58	8.06	20.36
Turkiye	88.39	4.64	6.97
Colombia	76.35	9.19	14.46
Portugal	78.24	10.73	11.04
New Zealand	67.05	9.81	23.14

When the details on a country basis are examined, it is seen that the only two countries in this group where the number of incoming researchers exceeds the number of outgoing researchers every year are the Netherlands and Portugal. These two countries also have the lowest inflation rates in this group.

Turkiye is the country that until the last 3 years has had more incoming researchers than outgoing researchers although it has the highest inflation rates throughout all years. The reason could be thanks to its geographical location being a bridge between Asia and Europe.

New Zealand has the lowest number of incoming and outgoing researchers in this group for all years. In addition, the ratio of incoming to outgoing researchers decreases every year. No clear economic reason has been found for this situation. However, the difficulty of transportation due to its geographical location can be suggested as the reason.

It was seen that the ratio of the number of incoming researchers to the number of outgoing researchers in Portugal was balanced and that the brain drain of the country could not be reconciled with economic or social reasons.

Destination Trends in Brain Drain. Turkiye and the Netherlands contribute the most brain drain to the United States and the United Kingdom. Portugal sends the majority of their brain drain to the United Kingdom and Spain. Having similar patterns to Portugal, Colombia directs its highest levels of brain drain to Spain and the United States. And the primary destinations for brain drain from New Zealand are Australia and the United States.

As seen in the previous groups and supported by Fig. 4, the two most preferred countries for brain drain in this group are the United Kingdom and the United States. Geographic location is seen to affect the preferred country for brain drain for New Zealand. Language similarity is also seen to affect the brain drain for Portugal and Colombia.

4 Discussion

The results of this study point to several critical perspectives on the brain drain phenomenon. The fact that there is an inverse proportion between high GDP values and brain drain up to a certain value and that there is a direct proportion between the inflation value and brain drain value in many countries, reveals the effect of economic factors on brain drain.

Another important finding is the effect of the language used on brain drain. The official language of the two countries that receive the most brain drain, the United States and the United Kingdom, is English. The language commonly used for international communication is also English. In addition, it was seen that Spain, Colombia, Portugal, and Brazil formed a brain drain network among themselves. The effect of this is the similar languages used. This finding shows the importance of social reasons for brain drain.

Finally, the effect of geographical location on brain drain was found. It was seen that the majority of the countries that were not in the European continent gave the most brain drain to a country that was geographically close to them.

5 Conclusion

This paper obtained a brain drain profile from researcher profiles using publicly available ORCID data. The results obtained were evaluated and interpreted. To better understand the results, the World Bank database was used, and a link was found between brain drain and economic reasons.

This study sheds light on the complex dynamics of brain drain, emphasizing the importance of cultural and social factors in addition to economic factors. Exploration into the dynamics of brain drain in the academic community has revealed intriguing patterns and insights. The analysis of researchers' movements across institutions and countries has unveiled correlations that illuminate the complex nature of knowledge migration.

In summary, the findings contribute valuable insights to the ongoing discourse on intellectual migration, hopefully opening avenues for further exploration within the academic community.

References

1. Docquier, F., Rapoport, H.: Globalization, brain drain, and development. J. Econ. Lit. **50**(3), 681–730 (2012)
2. Carr, S.C., Inkson, K., Thorn, K.: From global careers to talent flow: reinterpreting 'brain drain'. J. World Bus. **40**(4), 386–398 (2005)
3. Beine, M., Docquier, F., Ozden, Ç.: Diaspora effects in international migration: key questions and methodological issues. Swiss J. Econ. Stat. **146**, 639–659 (2010)
4. Giannoccolo, P.: The brain drain: a survey of the literature. Università degli Studi di Milano-Bicocca, Department of Statistics, Working Paper, (2006-03), p. 02 (2009)

Physical Activity and Mental Health: A Comparative Analysis Across Age Groups Using Complex Networks

Rama Krishna Thelagathoti[1](✉), Ka-Chun Siu[2], and Hesham H. Ali[1]

[1] University of Nebraska Omaha, Omaha, NE 68182, USA
{rthelagathoti,hali}@unomaha.edu
[2] University of Nebraska Medical Center, Omaha, NE 68198, USA
kcsiu@unmc.edu

Abstract. Physical activity is a crucial determinant of overall health and well-being, significantly influencing both physical and mental health outcomes. Understanding the intricate patterns of physical activity across different age groups and health conditions is essential for developing targeted interventions. Mental health conditions like depression and schizophrenia are known to affect physical activity levels, yet the extent of this impact across age groups remains under-explored. This study uses a complex network approach to investigate the influence of age and mental health conditions on physical activity patterns. We analyzed the relationship between physical activity and age in individuals with depression and schizophrenia, comparing these patterns with those in healthy individuals. Specifically, the study examines whether physical activity decreases with age and explores the correlations between physical activity levels and health status.

Keywords: Physical Activity · Mental Health · Complex Network Analysis · Mobility Data

1 Introduction

Physical activity plays a vital role in maintaining overall health and well-being. Physical activity is the ability to move freely and easily, is a critical aspect of daily life and is closely associated with overall health and well-being. It encompasses any movement that expends energy, ranging from daily activities such as walking and household chores to structured exercise and sports [3]. Regular physical activity is associated with numerous health benefits, including reduced risks of chronic diseases such as cardiovascular disease, diabetes, and certain cancers, as well as improved mental health and cognitive function [17] [13]. However, physical activity levels tend to vary across different age groups. Younger individuals generally engage in more physical activity compared to older adults, who may experience a decline in activity levels due to factors such as physical limitations, health conditions, and lifestyle changes [16].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 304–315, 2025.
https://doi.org/10.1007/978-3-031-82439-5_25

Physical activity is essential for maintaining overall health and well-being across all age groups, but the patterns and benefits of physical activity can vary significantly among young, middle-aged, and older individuals. Young adults typically engage in more vigorous and diverse physical activities, such as sports, running, and high-intensity interval training. This age group benefits from a relatively high metabolic rate and greater muscle mass, which supports their ability to perform intense physical activities [15]. However, the transition to higher education or full-time employment can lead to a decline in physical activity levels, highlighting the need for targeted interventions to sustain active lifestyles during this life stage. Middle-aged individuals often experience a shift in physical activity patterns due to increased work and family responsibilities. As people enter older adulthood, physical activity patterns tend to become less vigorous and more focused on maintaining mobility and functional independence [8].

Physical activity levels are significantly affected in individuals with mental health conditions such as depression and schizophrenia. These conditions are often associated with reduced motivation, fatigue, and physical health problems, all of which can contribute to lower levels of physical activity [14]. For instance, individuals with depression may experience symptoms such as anhedonia (loss of interest in activities) and psychomotor retardation (slowing of physical movement), which can severely limit their engagement in physical activities [7]. Similarly, those with schizophrenia may face challenges such as medication side effects, negative symptoms, and cognitive impairments, further reducing their physical activity levels [12].

This study aims to address several research questions related to physical activity patterns across different age groups and health conditions by leveraging complex network analysis. Following are the specific study goals of this manuscript.

- **Goal 1:** Investigate the Relationship Between Age and Physical Activity Levels: To determine whether physical activity levels decrease with age.
- **Goal 2:** Examine Physical Activity Levels Among Individuals with Depression and Schizophrenia: To explore the correlation between physical activity levels and health status among individuals with depression and schizophrenia, compared to a healthy control population.
- **Goal 3:** Explore the Correlation Between Physical Activity and Overall Health: To investigate the relationship between physical activity levels and overall health status among different age groups and health conditions.
- **Goal 4:** Identify Differences in Physical Activity Patterns Across Age and Health Groups: analyze and compare physical activity patterns among young, middle-aged, and older adults, and between healthy individuals and those with mental health conditions.

By utilizing a public dataset of sensor data collected over nine days from a diverse group of human subjects, this research employs correlation network model to analyze and compare physical activity patterns. Rest of the manuscript is organized as follows: Sect. 2 describes detailed methodology whereas Sect. 3

presents obtained results. Sections 4 and 5 provides the discussion of obtained results and limitations.

2 Methodology

In this study, we utilize complex network models to explore the correlation between mental health and physical activity. Complex networks are graphs that are used to model systems in certain application domains and do possess useful graph theoretic properties [2]. Complex network models are particularly effective in capturing dynamic interactions within systems, allowing for a deeper understanding of how changes in one part of the network can influence others [9]. This study's methodology is divided into several key steps: data acquisition, preprocessing and feature extraction, complex network representation, community detection, and community analysis. Each step is crucial for understanding the relationship between age, physical activity, and health conditions.

2.1 Data Set Description

The dataset used in this study was created by merging two independent datasets published by the same research group. The first dataset, 'Depresjon,' contains motor activity data from 55 adults: 23 with unipolar or bipolar depressive disorder and 32 healthy individuals [4]. The second dataset, 'Psykose,' includes 22 subjects diagnosed with schizophrenia and 32 healthy subjects. The healthy participants in both datasets are the same individuals [5]. By combining these datasets, a new dataset was generated, comprising three groups: depressive, schizophrenic, and healthy subjects. Depressive and schizophrenic subjects are considered the condition group, while healthy participants serve as the control group.

All participants wore a body-worn wearable sensor embedded in an Actigraph watch (Model: Actiwatch, Manufacturer: Cambridge Neurotechnology Ltd, England) [6]. The Actigraph watch, equipped with a piezoelectric accelerometer, assesses the intensity of limb movements, recorded as activity counts, with higher counts indicating more intense movements. Motion data was captured at a sampling frequency of 32 Hz and movements over 0.05g per minute. Participants wore the Actigraph watch on their right wrist for convenience, without any specific patterns or procedures, allowing natural mobility data collection over several days. Each participant provided motor activity data for 13 days, although some provided data for fewer days.

2.2 Preprocessing and Feature Extraction

Each participant's mobility data was stored in individual CSV files, recording motor activity at 1-minute intervals over 13 days. However, since not all participants had complete data, we used the minimum available 10 days, excluding the partial first day to avoid bias, resulting in 9 days of data per participant. The data was normalized and outliers removed. To capture differences in movement

intensity, we segmented the motor data into 24-hour intervals, calculating mean and standard deviation (SD) for each hour, producing 48 features. Day-wise averages and SDs added 18 more features, totaling 66. To examine the relationship between age, physical activity, and mental health conditions, we categorized the participants into three age groups as shown in Table 1: Young (20–34 years), Middle-aged (35–49 years), and Older (50–69 years). We selected 72 subjects who had at least 9 days of sensor data available. This preprocessing ensures a consistent analysis of motor activity.

Table 1. Age category distribution

Age category	Age Range	Count
Young Adults	24–34 years	21
Middle-aged Adults	35–49 years	29
Over50 Adults	50–69 years	22
Total		72

2.3 Complex Network Representation

Representing sensor data as a complex network offers significant advantages in understanding relationships and patterns within the data. In these networks, nodes represent individual subjects, and edges signify significant correlations in physical activity patterns, providing a visual representation that uncovers patterns and connections not apparent in raw data. The complex network is constructed based on two key properties: the degree of similarity in physical activity patterns and the alignment of network topology using community structure. Initially, the network is built using pair-wise correlations as a measure of distance between subjects, which helps identify strongly connected communities, thereby revealing intricate relationships within the subject population. Subsequently, network-based community detection methods, such as the Louvain clustering algorithm, are applied to detect clusters of subjects with similar physical activity patterns, aiding in understanding variations across different age groups and health conditions, and making the analysis more focused and meaningful [10,11].

The steps involved in this process are illustrated below.

Preliminaries. The methodology for creating the correlation network involves several assumptions:

- Assumption 1: N represents the total number of subjects included in the study.
- Assumption 2: K denotes the number of proposed features utilized to construct the correlation network, specifically 48-hour-wise features.

- Assumption 3: P_i and P_j represent randomly selected individual subjects from the population of N subjects. The indices (i, j) satisfy the condition $(i, j) \leq N$ and $(i, j) > 0$, ensuring valid subject indices within the population.
- Assumption 4: The value $\rho[i, j]$ corresponds to the Pearson pairwise correlation coefficient between subjects i and j. CM refers to the Correlation Matrix generated by computing $\rho[i, j]$ for all valid subject pairs (i, j). In other words, $CM[i, j]$ contains the correlation coefficient $\rho[i, j]$ between subjects i and j.
- Assumption 5: T represents a predefined threshold set by the user to determine the minimum strength of correlation required for constructing the complex network. Generally, a correlation coefficient above 0.5 is considered significant.
- Assumption 6: SM denotes the significance matrix obtained after applying the threshold T to the correlation coefficients in CM. SM serves as an adjacency matrix, where $SM[i, j]$ takes the value 1 if the correlation between subjects P_i and P_j exceeds the threshold T, and 0 otherwise.

Procedure. To create a complex network representing the similarity between participants based on their mobility patterns, the following steps are undertaken:

- Compute Pair-wise Pearson Correlation:
  - Calculate the Pearson pairwise correlation coefficient ρ between each pair of subjects P_i and P_j, where i and j range from 1 to N.
  - $\rho[i, j]$ represents the correlation coefficient between subjects P_i and P_j.
- Generate Correlation Matrix (CM):
  - Create a correlation matrix (CM) of size $N \times N$, where N is the number of subjects in the study.
  - Each element $CM[i, j]$ of the matrix represents the ρ value between subjects P_i and P_j.
- Set Threshold (T) for Correlation Strength:
  - Choose a predefined threshold value T to determine the strength of the correlation required for constructing the complex correlation correlation.
  - Typically, a threshold above 0.5 is considered strong, but the specific value can be adjusted based on the desired criteria.
- Generate Significance Matrix (SM):
  - Create a significance matrix (SM) based on the correlation values in CM using the formula:

$$SM[i, j] = \begin{cases} 1 & \text{if } CM(P_i, P_j) \geq T \\ 0 & \text{otherwise} \end{cases}$$

  - SM is an adjacency matrix, where each element $SM[i, j]$ determines whether an edge should be present between subjects P_i and P_j in the final network.
- Generate Complex Network:
  - Based on the values in the SM matrix, generate the complex network.

- For each pair of subjects (P_i, P_j) where $SM[i, j] > T$ (indicating a significant correlation), create an edge connecting P_i and P_j in the network.

Each participant is represented as a node, and the edges represent the similarity between their mobility patterns based on the correlation threshold. By following these steps, a network is created, providing a visual representation of the strength of correlation between participants' mobility patterns.

2.4 Community Detection and Analysis

After constructing the complex network, we employ the Louvain clustering algorithm to identify strongly connected clusters within the network. This widely used method for community detection efficiently uncovers high-modularity partitions by iteratively optimizing the network's modularity [1]. In this network, nodes represent participants, and edges signify significant correlations in their physical activity patterns. The Louvain algorithm helps group participants with similar activity patterns, thereby elucidating how physical activity is distributed among different age groups and health conditions. Community analysis follows as a critical step in this study. It involves examining the dominant clinical characteristics and common demographics within each identified cluster to understand the reasons behind their formation. This analysis also facilitates the visualization of physical activity patterns specific to each cluster, revealing distinct differences and similarities.

3 Results

This section presents the results of our analysis, categorized by age groups: young adults (20–34 years), middle-aged adults (35–49 years), and older adults (50–69 years). The findings are illustrated through network representations and bar plots, highlighting the dominant clinical attributes in each category and the physical activity patterns across condition and control groups. Figures 1(a), 1(c), and 1(e) show the network representations of the three age categories: young, middle-aged, and older adults, respectively. Each node in these networks represents an individual participant, and the edges indicate significant correlations in their physical activity patterns. The nodes are color-coded based on group and community membership. To recognize group membership, control group subjects are preceded by 'c', depressed group subjects by 'd', and schizophrenic group subjects by 's'. Similarly, to recognize community membership coloring scheme is used. For example, in Fig. 1(a), subjects c5, c23, and c25 form community Y1, which includes participants from all three groups—control, depressed, and schizophrenic—as illustrated in Fig. 1(b).

3.1 Dominant Clinical Attributes

Figures 1(b), 1(d), and 1(e) present bar plots illustrating the dominant clinical attributes observed within each age category. These visualizations depict the

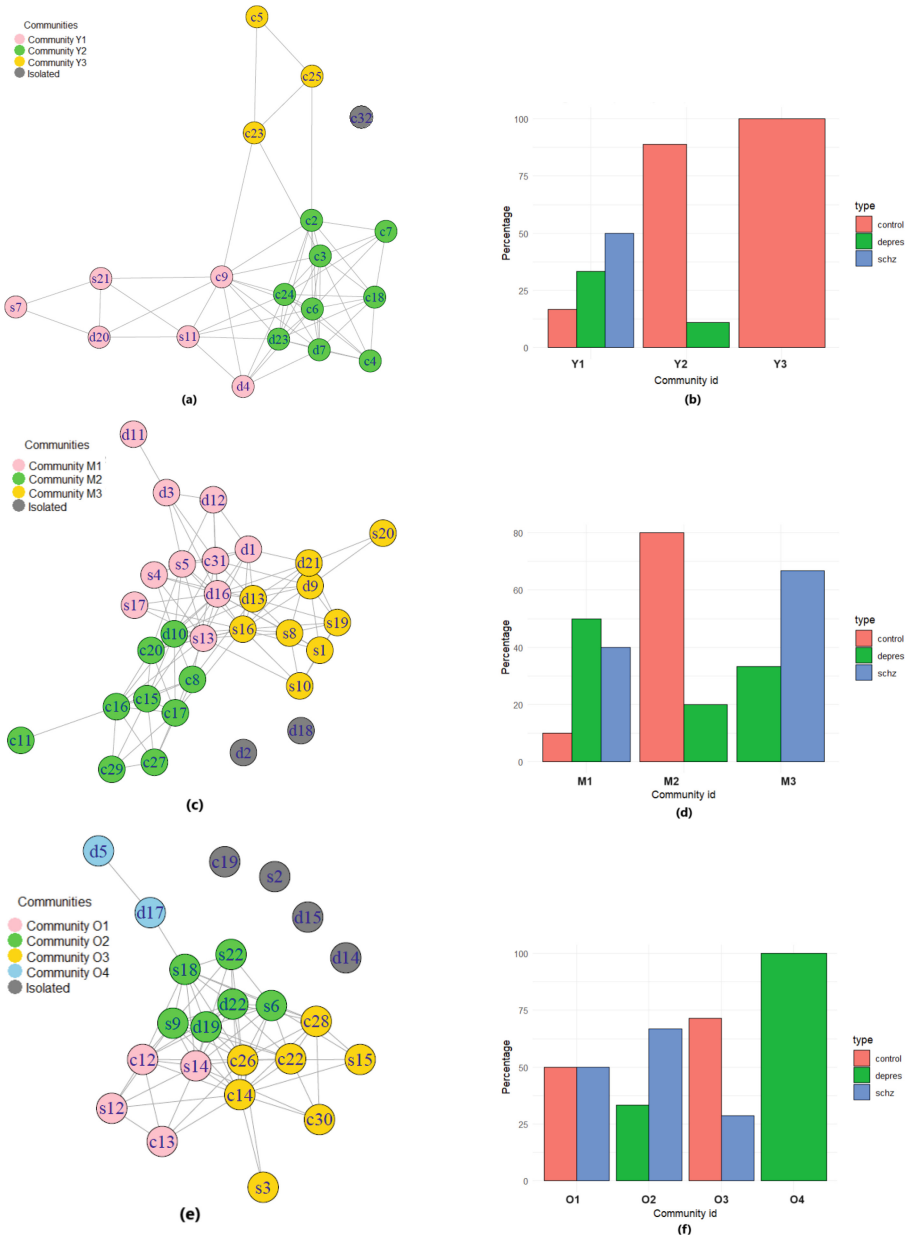


Fig. 1. (a) Young Adults (24–34 years): Network representation (b) Dominant clinical parameters (c) Middle aged (35–49 years): Network representation (d) Dominant clinical parameters (e) Over50 Adults (50–69 years): Network representation (f) Dominant clinical parameters

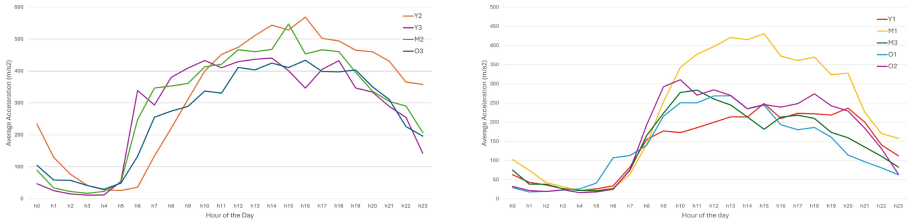


Fig. 2. Over50 Adults (50–69 years): Network representation (left) Dominant clinical parameters (right)

significant clinical attributes associated with each community identified across the age groups.

In the young adults group (20–34 years):

- Community 3 comprises entirely of subjects from the control group.
- Community 2 includes 90% subjects from the control group.
- Community 1 predominantly consists of condition group subjects, encompassing both individuals with depression and schizophrenia.

In the middle-aged adults group (35–49 years):

- Community 3 consists entirely of subjects from the control group.
- Communities 1 and 2 are predominantly composed of condition group subjects.

In the older adults group (50–69 years):

- Community 4 comprises entirely of subjects from the control group.
- Community 2 is entirely composed of condition group subjects.

These findings suggest that subjects from the control group tend to form distinct communities, whereas those from the condition groups (depression and schizophrenia) tend to cluster together. This clustering underscores the divergence in physical activity patterns between these distinct groups.

3.2 Physical Activity Patterns

The two plots in Fig. 2 shows the average acceleration (m/s^2) of physical activity across a 24-hour period for different subject groups. The first plot represents data from the control dominant groups across different age groups (healthy individuals), while the second plot shows data from subjects with conditions such as depression and schizophrenia.

The analysis reveals several key insights:

- **Peak Activity Levels:** Control group subjects (Y2, Y3) exhibit higher peak activity levels than condition group subjects (Y1), indicating more vigorous daily routines and better physical health among healthy young adults. This trend is consistent across all age categories.
- **Daily Activity Trends:** Both groups show increased activity starting in the morning and peaking in the afternoon. However, the control group maintains higher accelerations and sustains these levels longer into the afternoon compared to the condition group.
- **Activity Decline:** The decline in physical activity starts earlier and is more pronounced in the condition group, especially noticeable in middle-aged (M1, M3) and older adults (O2). This could indicate the impact of mental health conditions on overall physical activity levels.
- **Young Adults:** The control group's young adults (Y2, Y3) are highly active compared to the condition group's young adults (Y1), reflecting the impact of depression and schizophrenia on reducing physical activity.
- **Middle-aged Adults:** Control middle-aged adults (M2) show robust activity patterns, whereas the condition group's middle-aged adults (M1, M3) exhibit significantly lower activity levels, suggesting a considerable impact of mental health conditions.
- **Older Adults:** Control older adults in group O3 maintain moderate activity levels, while the condition group's older adults (O1, O2) show markedly lower activity, indicating a more significant decline in physical activity with age and mental health conditions.

The comparative analysis highlights significant differences in physical activity patterns between control and condition groups across various age categories. Control group subjects exhibit higher and more consistent activity levels, suggesting better physical health and daily routines. In contrast, the condition group shows reduced and less variable activity, reflecting the negative impact of mental health conditions such as depression and schizophrenia. These findings emphasize the need for tailored interventions to enhance physical activity among individuals with mental health conditions, aiming to improve both physical and mental health outcomes.

3.3 Implications

From the analysis, the following conclusions can be drawn:

- *Age and Mobility:* There is no significant degradation in mobility with age among control group subjects. This implies that maintaining good physical health can mitigate the effects of aging on mobility, highlighting the importance of regular physical activity and a healthy lifestyle.
- *Lower Mobility in Condition Groups:* Condition group subjects exhibit lower mobility levels compared to the control group. This reduced mobility underscores the physical manifestations of depression and schizophrenia, emphasizing the need for targeted physical activity interventions to improve the overall health of these individuals.

- *Similar Sleep-Wake Patterns in Condition Groups:* Depressed and schizophrenic groups have similar sleep-wake patterns, which is not observed in the control group. This similarity points to a potential disruption in circadian rhythms associated with these mental health conditions, suggesting that interventions aimed at regularizing sleep patterns could be beneficial.
- *Tailored Interventions for Mental Health:* The lower mobility levels and similar sleep-wake patterns in depressed and schizophrenic groups suggest a need for tailored interventions. Programs aimed at improving physical activity and regularizing sleep patterns could significantly enhance the quality of life for individuals with these conditions.
- *Early Detection and Monitoring:* The network-based correlation network model can be utilized for early detection and continuous monitoring of mental health conditions. By identifying individuals whose physical activity patterns deviate from the norm, healthcare providers can intervene earlier, potentially mitigating the severity of the conditions.
- *Public Health Strategies:* These findings can inform public health strategies aimed at promoting physical activity across all age groups. By emphasizing the benefits of maintaining high physical activity levels, even in older adults, public health campaigns can encourage lifelong habits that contribute to better health outcomes.

These findings underscore the impact of mental health conditions on physical activity and highlight the importance of targeted interventions to improve physical activity levels among individuals with depression and schizophrenia.

4 Discussion

The network analysis for each age category uncovers intriguing patterns and outliers in the physical activity data. Some nodes are disconnected, representing individuals with movement patterns significantly different from their peers. These isolated nodes, considered outliers, exhibit unique daily routines, movement accelerations, or sleep activities. For instance, in Fig. 1(a), node C32 corresponds to a control group subject whose activity patterns are distinct from others in their age group, highlighting the variability within the population.

Furthermore, the network reveals that certain subjects with health conditions cluster within communities dominated by other health conditions or the control group. In Fig. 1(a), Group Y2 is mainly composed of control group members but includes d23, a subject diagnosed with depression. Interestingly, d23's mobility patterns align more with the control group than with other depressed individuals, indicating that their physical activity is less impaired than expected for someone with depression. This blending of condition group subjects into control communities suggests variability in the effects of mental health conditions on physical activity. Some individuals, despite clinical diagnoses, may maintain mobility patterns similar to healthy peers, highlighting the importance of considering individual variability in such analyses.

One key benefit of this approach is its potential to support personalized interventions based on mobility patterns. By identifying deviations from typical movement behaviors, healthcare providers can create tailored treatment plans that address individual needs, leading to better treatment outcomes and more efficient resource allocation. Additionally, mobility pattern insights can guide community-specific mental health programs. Understanding trends within populations allows for targeted public health initiatives that address the distinct challenges of different communities, enhancing the effectiveness of interventions.

In summary, these findings underscore the complexity of physical activity patterns across various age groups and health conditions. The presence of outliers and the integration of condition group subjects into control communities demonstrate that individual differences are significant. Personalized approaches in interventions are essential to improving physical activity levels in individuals with mental health conditions.

5 Limitations and Future Directions

This study has limitations, including a small, non-diverse sample, inconsistent sensor data quality, and reliance on self-reported information, which may affect generalizability. The focus on physical activity alone, without considering other health metrics like heart rate or sleep quality, limits the analysis. The cross-sectional design prevents causal inferences, suggesting the need for longitudinal studies. Future research should incorporate multimodal data, advanced machine learning, and user-centered designs to improve health monitoring. Expanding to other health conditions and collaborating with policymakers could enhance public health programs promoting physical and mental well-being.

6 Conclusion

In this study, we conducted a comparative analysis using a complex network model to examine the influence of mental health conditions on physical activity patterns across different age groups. We analyzed the relationship between physical activity and age in two groups: one comprised of individuals diagnosed with depression and schizophrenia, and the other of healthy individuals. A key research question was whether the decline in physical activity follows a different pattern between the two groups. The results highlight significant differences in physical activity patterns across age groups and health conditions. Notably, while physical activity levels remain relatively stable with age among healthy individuals, those with mental health conditions, particularly depression and schizophrenia, show a marked decline in activity as they age. This underscores the impact of mental health on aging and physical mobility, suggesting that activity patterns could serve as indicators of mental health status.

References

1. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mecha. Theory Expe. **2008**(10), P10,008 (2008)
2. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: structure and dynamics. Phys. Rep. **424**(4–5), 175–308 (2006)
3. Caspersen, C.J., Powell, K.E., Christenson, G.M.: Physical activity, exercise, and physical fitness: definitions and distinctions for health-related research. Public Health Rep. **100**(2), 126 (1985)
4. Garcia-Ceja, E., et al.: Depresjon: a motor activity database of depression episodes in unipolar and bipolar patients. In: Proceedings of the 9th ACM Multimedia Systems Conference, pp. 472–477 (2018)
5. Jakobsen, P., et al.: Psykose: a motor activity database of patients with schizophrenia. In: 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS), pp. 303–308. IEEE (2020)
6. Ltd, C., Inc, C.: Camntech (2022). https://www.camntech.com/. Accessed 17 July 2024
7. Mura, G., Carta, M.G.: Physical activity in depressed elderly. a systematic review. Clin. Pract. Epidemiol. Mental Health: CP & EMH **9**, 125 (2013)
8. Nelson, M.E., et al.: Physical activity and public health in older adults: recommendation from the American college of sports medicine and the american heart association. Circulation **116**(9), 1094 (2007)
9. Newman, M.: Networks. Oxford university press, Oxford (2018)
10. Newman, M.E.: The structure and function of complex networks. SIAM Rev. **45**(2), 167–256 (2003)
11. Pósfai, M., Barabási, A.L.: Network science. Citeseer (2016)
12. Rosenbaum, S., Tiedemann, A., Sherrington, C., Curtis, J., Ward, P.B.: Physical activity interventions for people with mental illness: a systematic review and meta-analysis. J. Clin. Psychiatry **75**(9), 14465 (2014)
13. Saxena, S., Van Ommeren, M., Tang, K., Armstrong, T.: Mental health benefits of physical activity. J. Ment. Health **14**(5), 445–451 (2005)
14. Schuch, F.B., et al.: Physical activity and incident depression: a meta-analysis of prospective cohort studies. Am. J. Psychiatry **175**(7), 631–648 (2018)
15. Strong, W.B., et al.: Evidence based physical activity for school-age youth. J. Pediatr. **146**(6), 732–737 (2005)
16. Sun, F., Norman, I.J., While, A.E.: Physical activity in older people: a systematic review. BMC Public Health **13**, 1–17 (2013)
17. Warburton, D.E., Nicol, C.W., Bredin, S.S.: Health benefits of physical activity: the evidence. CMAJ **174**(6), 801–809 (2006)

Enhanced Vehicle Routing Algorithm for Reinforcement Learning Using Artificial Potentials for Multi-objective Logistic Network

Taku Iguchi[1(✉)], Kei Uchimura[1], Shigeyuki Miyagi[1,2], and Osamu Sakai[1,2]

[1] Department of Electronic Systems Engineering, The University of Shiga Prefecture, Hassaka-cho 2500, Hikone, Shiga 522-8533, Japan
on23tiguchi@ec.usp.ac.jp

[2] Regional ICT Research Center for Human, Industry and Future, The University of Shiga Prefecture, Hassaka-cho 2500, Hikone, Shiga 522-8533, Japan

Abstract. Logistic routing problems inherently include mathematical difficulties in solution optimization and also strongly depend on specific load distributions in path network. In this study, to overcome these difficulties, we introduce artificial potentials in network topology in an environment where uncollected and undelivered goods are located in nodes with goods currently carried by agents. This inclusion of the hypothetical physics-informed effects in our reinforcement model enhances rapid convergence of logistic route solution, whose optimization level is confirmed by higher-order Markov chain networks. This method also works in cases where a network includes edges with different weights like highway effects, close to specific calculation environments along with real-world demands.

Keywords: Route planning · Markov chain · Potential field · Reinforcement learning

1 Introduction

Vehicle routing problems (VRPs) [1,2], which is an advanced task from traveling salesman problems, include an inherent significant problem with non-deterministic polynomial (NP) hardness [1,3], where required calculation time becomes long in power-law scale as distance of the overall route increases. A general logistic route environment, in which vehicles for freight transport move among pickup and delivery locations, is complicated as VRP tasks. Specific parameters like count of goods types, agent number and network topology with pickup and delivery locations vastly differ in given cases. So far, various route planning methods such as reinforcement learning (RL) models [4–7] and heuristic approaches [3,8,9] have been proposed, without any universal tool that is effective in any cases.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 316–327, 2025.
https://doi.org/10.1007/978-3-031-82439-5_26

We focus on route planning by RL in this study. RL algorithms are well examined for multiple purposes, in which, to our references, logistic planning and VRP [5,7,10] and maze-solving tasks [11–14] have been listed up as previous research achievements. Following these previous studies, we also investigated multi-agent simulations based on action performances and state observations [15]. Another related result in our group is maze-solver for a modified RL model [16], mimicking experimental conditions of high-voltage potentials and mobilities in charged particles.

To upgrade RL models for rapid calculations of logistic routes, in this study, we propose to set artificial potentials [17,18] in a calculation environment to represent spatial locations of uncollected and undelivered loads with those currently carried by agents. Such artificial potentials were successfully applied to robot navigation system, where target location has positive potential while obstacles possess negative potentials. Movable robots in *physical* space sense such artificial potential fields, and avoiding negative potential regions, they are attracted by positive potentials, like negatively-charged particles move forward to positive potential areas. When we include such hypothetical potentials in our *network* space, it is unclear that agents in our logistic model would be remotely but sensitively activated by such artificial potential fields. In our previous study about maze-solving, the numerical code based on modified RL model successfully handles hypothetical potential in which one targeted node is fixed in a simple and small network space [16], although the network in our logistic model is too complicated to simplify functions of hypothetical potentials on route findings. Thus, the crucial point we have to clarify in this study is whether artificial potential fields are well synthesized with network topology and RL algorithm, in particular, initially-installed rewards r and expected discounted rewards (or action values) Q that is derived after iterative calculation based on r. What's more, we consider insertion of shortcut paths in a regular-lattice network, resembling geographical ones, and we attempt to confirm validities of our proposed method for networks with the so-called small-world topology [19].

Then, to confirm route convergence around the optimal solution, large-size complex networks in Markov chains (MCs) play crucial roles [20–22]. Since problems are with NP hardness, it is also tough estimation whether routes in findings are close to optimum states or not. In our previous studies [15], we visualized higher-order MCs with the order up to four, whose node indicates a current location with memories of the previously visited three nodes. Then, even if geographical nodes are limited to 25, its fourth MC network possesses 912 nodes and 3,132 edges. Then, entangled routes in the geographical five-by-five lattice network are clearly disentangled as simple traces in the fourth MC network.

In this study, we describe an RL-model algorithm based on artificial potential fields for sensing varying status of pickup and delivery locations, and rapid convergence of route solutions are confirmed using high-order MC networks. Comparison between our model and a typical Q-learning method verifies rapid and drastic enhancements in convergence by artificial potentials, and some applicable merits to flexible adaptation to cases of highway-like routes are also demon-

strated. In Sect. 2, we describe details of our model, and Sect. 3 displays main calculation results in many spatial patterns. In Sect. 4, traces in higher-order MC networks are visualized, which is followed by Conclusion in Sect. 5.

2 Model and Method

2.1 Artificial Potential Field Model

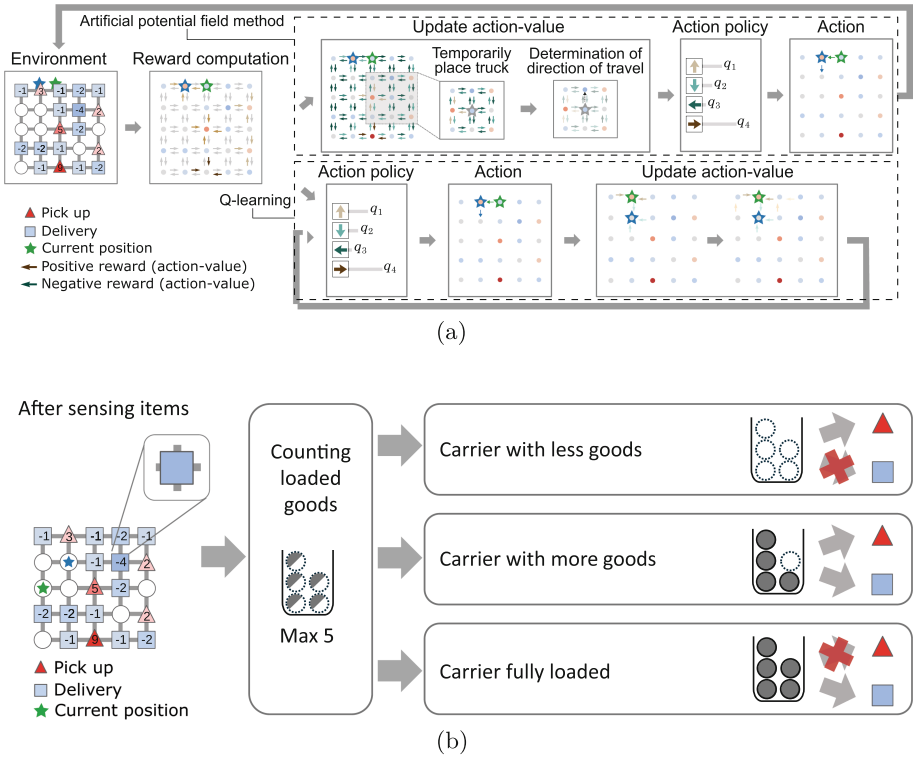


Fig. 1. (a) Schematic views of flow chart of artificial potential field model in comparison with classical RL model. (b) Policies of actions depending on rate of vacancy in loading area. Setting of reward polarities corresponds to these policies.

Our model proposed in this study is based on a conventional Q-learning scheme in RL. After reviewing the Q-learning form briefly, we describe details of our model, in which some elements are additional and supplementary for our purpose: logistic VRPs. We note that we restrict our interests in virtual agents mobile in networks with nodes and edges, where agents can move on edges in both directions, and the optimal guide of agents is the principle we aim at finding (Fig. 1).

A typical Q-learning scheme which was applied to general maze-solving problems [11–14] is described as follows. One sets positive r to edges linked to the goal or exit node, with small negative r to other edges. Then, agents aim at reaching goal to get r in a shorter path, avoiding extra length in the route. While such r setting is fixed over iterations i, a Q profile is calculated by the equation:

$$Q_i\left(s_n, a_n\right) = \left(1-\alpha\right) Q_{i-1}\left(s_n, a_n\right) + \alpha\left[r_n + \gamma \max_{n'} Q_{i-1}\left(s_{n'}, a_{n'}\right)\right], \quad (1)$$

where n and n' are the current and next possible nodes, and a_n is *action*, which means edge selection as the next path in maze-solving. s_n implies *state*, implying pursuing or reaching the goal. *Learning rate* α, one of the two parameters one sets for calculation, is the rate of replacing previous Q by a new one. The other parameter is *discount rate* γ, meaning the rate of preference for future incomes, like $Q\left(s_{n'}, a_{n'}\right)$ at next possible action $a_{n'}$ or later. In our previous computation, we fix these values as: $\gamma = 0.9$ and $\alpha = 0.1$ [16]. After iterative numerical runs at different nodes by starting from the entry to the exit arrival, the Q profile along a route between the entry and the exit of the maze is updated. When we call a passage of this agent motion as an episode, an agent can find a shorter path as episodes increase. We calculate logistic routing by this original Q-learning calculation method as a comparative attempt in this study, as shown later.

To accelerate logistic route findings effectively, we apply additional and supplementary functions to the classical Q-learning method. Comparing with maze-solving tasks, we change the virtual environment of RL in the following items: (a) multiple entries and exits in an analogical maze scheme, (b) five states (s_n at node n) of agents which are for pursuing pickup locations (nodes) (sP1), reaching pickup nodes (sP2), pursuing delivery nodes (sD1), reaching delivery nodes (sD2), and pursuing both pickup and delivery nodes (sPD), and (c) multi-directional motions of agents, unlike the cases of maze solving in which paths and traces of agents are unidirectional between one pair of entry and exit.

To realize the item (b), we introduce artificial potential fields in this study. In our model, goods carried by an agent are counted at every step j of the agent action; if the carrier of an agent is almost empty, we set states of sP1 or sP2 with r positive and negative near pickup (r_{p}) and delivery (r_{d}) locations respectively, while they are sD1 or sD2 when the number of goods in the loading area is at or close to the capacity of an agent with r positive and negative near delivery and pickup locations respectively. The other is sPD, in which an agent carries some goods but the empty parts exist. According to changes in s_n, we set the corresponding actions a_n like: one step closer to pickup node(s) (aP1), arrival and loading at one pickup node (aP2), one step closer to delivery node(s) (aD1), arrival and unloading at one delivery node (aD2), and one step closer to pickup or delivery node(s)(aPD). Such behaviors resemble a physics informed model in which attractive and repulsive forces occur between positive/negative charges in positive/negative potentials. Of note, masking strategies in which inappropriate nodes are tentatively removed in algorithms are popular [10], but our scenario which handles positive/negative r is more progressive to take balance between pickup/delivery locations. Robotic navigation systems are based on such physical

models to avoid obstacles on their tracks, and following this concept in robotics [17,18], here we call this hypothetical mechanisms as artificial potentials. As a summary, the reward function $r(j, e_x)$ on an edge e_x at j-th step of the agent v with goods $l_v(t)$ is:

$$r\left(j, e_x\right) = \begin{cases} -r_{\mathrm{p}}\left(j, e_x\right) + r_{\mathrm{d}}\left(j, e_x\right) & \text{if } l_v\left(j\right) = l_{\max} \\ r_{\mathrm{p}}\left(j, e_x\right) + r_{\mathrm{d}}\left(j, e_x\right) & \text{if } q < l_v\left(j\right) < l_{\max} \\ r_{\mathrm{p}}\left(j, e_x\right) - r_{\mathrm{d}}\left(j, e_x\right) & \text{if } l_v\left(j\right) \leqq q\,, \end{cases} \tag{2}$$

where $l_{\max}$ is the maximum number of goods that one agent can carry and q represents maximum goods left in the environment.

In addition to this artificial potential installation, from the requirement in item (c), we replace iterative runs based on Eq.(1) by one-step calculation at *arbitrary* nodes and data updates and accumulated in such random attempts, and Q profiles are not biased around some parts but their updates are distributed uniformly. Accordingly, our model can work to support agent motions throughout the targeted logistic area.

2.2 Higher-Order Markov-Chain Network

MCs or Markovian domains are very popular concepts in various fields of information theory and RL, and Q-learning itself is suitable for guides of agents acting optimally in controlled Markovian domains [20–24]. Then, if such guides are successful, MCs that agent traces will be expected to be on favorable routes for logistic VRPs. In our previous study [15], we propose higher-order MC networks to visualize optimized routes in logistic design [15], with capturing significance of statistical properties like scale-freeness in agent actions [25]. In this study, we derive traces of agents in higher-order MC networks to estimate levels of solution convergence.

In Sect. 4, we show networks of the first (simple) and the third-order MCs. The first MC network is the hypothetical geographical network itself, which is the 5×5 regular lattice network with nodes in this study. A node n_j in this network displays spatial location of an agent who visits it at a spatiotemporal move j. In the second-order MC network, one node represents information of agent two-sequential visits, i.e., the current location n_j and its one-step-before location n_{j-1}. In the third-order MC network, one network node represents data of a $n_{j-2} - n_{j-1} - n_j$ chain. That is, memory of previously-visited locations of hysteresis is included in the network topology.

3 Results

3.1 Route Findings And Comparison with Standard Q-Learning Method

We performed a number of calculations using our model in various types of networks. The numbers of agents is two, and goods up to five types are distributed initially and virtually carried by agents. To ensure validity universally

over many cases, initial installation and delivery demands are arbitrarily prepared as 10 sets in the following analysis, where five types of goods and 21–38 goods in total with arbitrary balances among types are serviced. To update Q by Eq. (1), we performed 400 iterations for Q profiles to select agent actions.

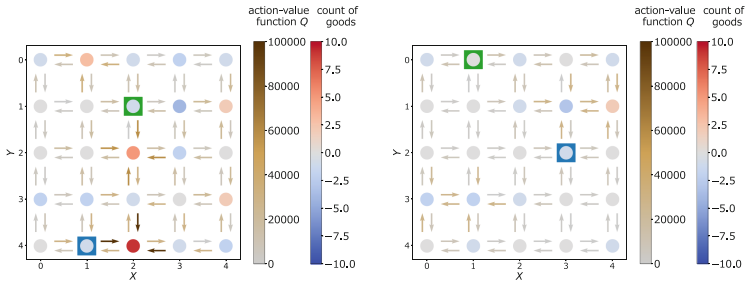


Fig. 2. Typical examples of Q distributions with remaining goods locations. Closed squares indicate current position of agents. Attached axes are hypothetical spatial coordinates along 5×5 square-lattice networks, where one scale unit is an network edge. (a) For initial setting ($j = 1$). (b) In in-between phase of computation ($j = 6$).

One of the advantages of our model is dynamic transition of Q profiles as logistic tasks like picking up and delivering goods are in progress. Figure 2 shows examples of Q profiles at the very beginning ($j = 1$) and in the middle ($j = 6$) phases of the total logistic process for one specific pickup/delivery case. Initially, the loading area of the agent is almost empty. s of the agent is at sP1, and r is positive around the pickup stops while r around delivery points are negative; no achievements will be expected in a delivery phase in cases without goods. Resultant Q has a peaking profile around positive r edges, and the agent senses its foot locally, proceeding to find more higher Q. In the meanwhile, if the agent carries some amount of goods in the in-between phase, r around both pickup and delivery locations becomes positive, with s = sPL; more goods will be loaded at a pickup site, and goods carried in the loading area will be unloaded at a delivery site. Then, around both sites, Q is in the higher levels.

Our model seems to be reasonable for such logistic task in which multiple pickup and delivery sites are spatially dispersed, and its advantage is confirmed by comparing outputs with those calculated by other methods. Figure 3 shows shortest routes found in model calculations as the count of episodes increases, where one episode corresponds to progress from the beginning to the completion of one logistic task by agents. When we use the classical Q-learning method, it takes tremendous episodes to reach optimized route. This is attributed to multi-objective logistics that is difficult to find a straightforward convergence of the solution. In contrast, our model finds a much shorter route from the very initial count of episodes; effects of artificial potential fields with random attempts for Q updates are quite powerful to solve such multiple pickup and delivery problems.

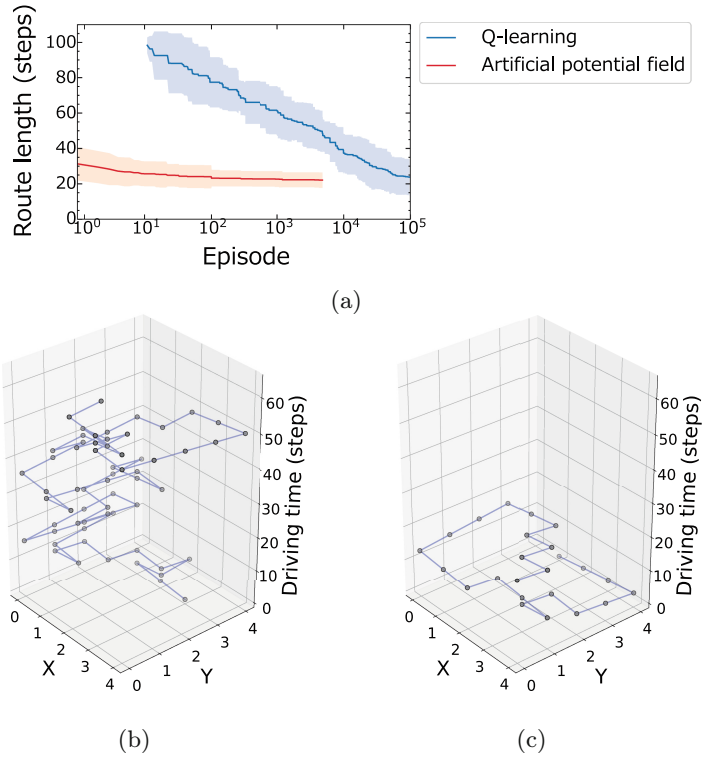


Fig. 3. Comparative computations by classical Q-learning and artificial potential field methods. (a) Minimum route lengths, measured by passing edges, over 10 logistic patterns as a function of number of calculated episodes. In cases of Q-learning, when episodes are less than 100, no routes whose lengths are less than 100 are found. (b) One typical example of derived route of one agent by Q-learning method at 1,000 episodes. (c) One typical example of derived route by artificial potential field method at 1,000 episodes.

Figure 3 displays typical route examples of an agent throughout total progress for one logistic task by the classical Q-learning and our methods. At one episode count (= 1,000), the route found by our method was much shorter than that observed in the Q-learning method. Artificial potential fields can guide our agent to show favorable choices several steps in advance of its current position. Another practical merit achieved in our method is shortening of computation time. Until this episode, the Q-learning method required approximately 1.3×10^4 s, whereas it took $\sim 8.3 \times 10^3$ s in our method; for one episode, only 0.8 s was required, which corresponds to almost 40% reduction for route finding. This fact implies artificial potential fields accelerate route planning effectively.

Size of networks handled in our artificial potential method is not limited in the range shown in this study. Beyond 9×9 networks, actually, we have already successfully obtained appropriate solutions for route planning for a network with

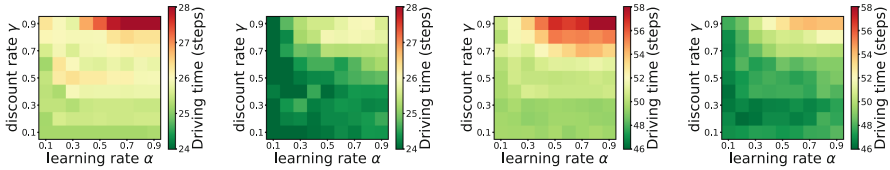
20×20 regular-lattice network (not shown). This size is equivalent to 20×20-km^2 area in a typical road map in Japan. Densities of pickup and delivery sites for local Japanese logistics are so high that this size corresponds to a daily task for 5–10 agents, which is within the capacity of our calculation model.

These facts indicate that the concept of artificial potentials is well matched with r and Q in RL models, with merits of short calculation times for route findings. The optimum levels of derived routes will be also estimated by large-size complex networks with high-order MCs in Sect. 4 (Fig. 6).

3.2 Effects of Shortcuts Inclusion in Logistic Routes

Table 1. Shortcut performances and their consequences on average path lengths, degree distributions and average minimum travel steps over overall time.

| size | shortcuts | average path length | nodes | | |



(a) (b) (c) (d)

Fig. 5. Overall driving times in units of steps on $\alpha - \gamma$ maps. Driving time has similar quantity to route length (counted by passing edge number) without shortcuts; at one time step, an agent moves to an adjacent node along one edge. On a shortcut edge, agents move for a longer route length as indicated geographically. (a) On 5×5 network without shortcuts. (b) On 5×5 network with shortcuts. (c) On 9×9 network without shortcuts. (d) On 9×9 network with shortcuts.

Eq. (1), possible contribution to Q by r is αr with edge-length effects, therefore this decrement in α where $0 < \alpha < 1$ is consistent with distance factors in Q. To investigate effects of such shortcuts, setting two different regular-lattice networks, we performed two-agent computation runs for 30 random pickup and delivery patterns with five-type goods.

Table 1 represents mean path lengths and other features of targeted networks with or without shortcuts. Apparent changes are found in degree distributions, and large degree nodes increase by insertion of shortcuts. In either cases of two sizes, shortcuts insertion makes the mean path lengths shorter by factors of 20–30%, which arequantitative changes caused by small-world effects. After calculations of routes by varying logistic patterns over all ranges of α and γ, mean minimum steps of agents for route findings is also reduced by shortcut insertion, which is net benefits for operation managers as well as for customers. In those computations, to update Q by Eq. (1), we performed 400 iterations for Q profiles of 5×5 networks and 1,296 iterations for those of 9×9 networks.

Figure 5 shows profiles of the shortest route lengths on the α-γ plane in cases of these two networks. Note that typical Q learning calculations for maze solving is performed in parameter sets with $\gamma \sim 0.9$ and $\alpha \sim 0.1$. We observed significant reduction of the total route length measured by the count of passing edges; an agent can use shortcuts effectively, and Q successfully spreads along them.

That is, these results verify that artificial potentials are also valid for a small-world network. In addition to real shortcut effects, they can work successfully also in this RL model.

4 Discussion

The results shown in Sect. 3 are comparatively and relatively true, but there is no information for optimization levels of the derived routes. In this section,

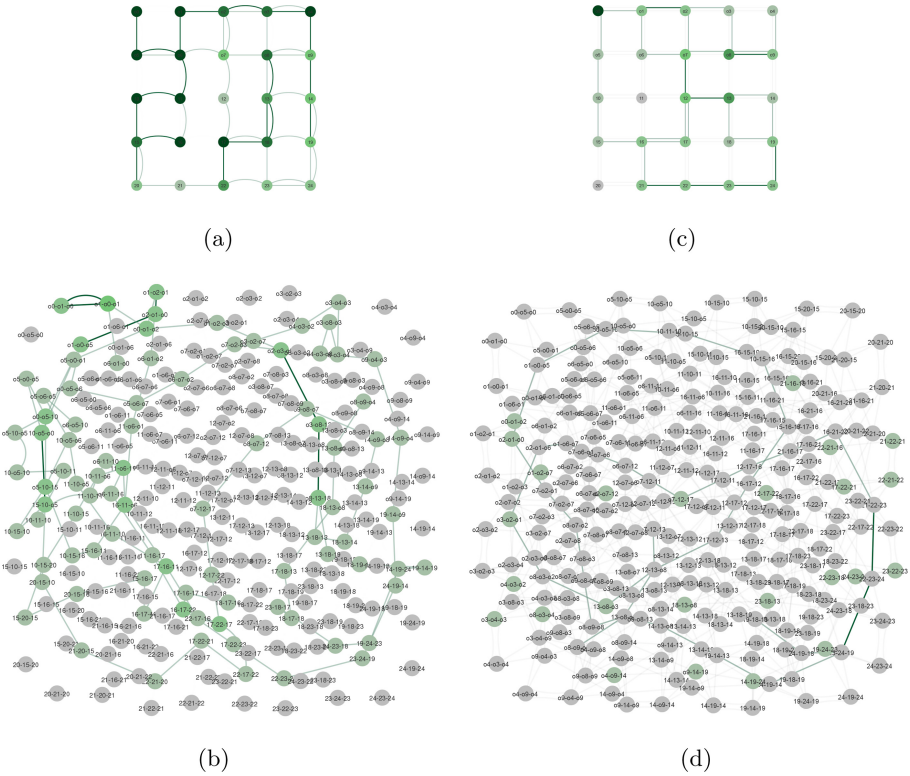


Fig. 6. Simple and third-order higher-order MC networks for visualization of routes which are plotted in Fig. 3. Darkness level of edge color are proportional to probability of agent choices. (a) Routes of two agents on simple MCs found in classical Q-learning method. (b) On third-order MCs found in classical Q-learning method. (c) On simple MCs found in artificial potential field method. (d) On third-order MCs found in artificial potential field method.

we examine them using high-order MC networks. As briefly reviewed in Sect. 2, MC networks are useful to check out optimized routes; if an optimized route is visualized inside such large-size complex networks, it will exhibit a quite smooth one-line trace without any branches and circular motions.

Figure 6 shows examples of the shortest routes (Figs. 6(c) and (d)) in one case of a specific pickup and delivery environment, in comparison with another case with a longer route (Figs. 6(a) and (b)). As the same trends reported in our previous study [15], we clearly observe two completely disentangled routes. In other words, a designed route will be in a higher optimized level, like in Fig. 6(d), which is the third-order MC network, if it is disentangled in a lower order MC network. In the case of Fig. 6(d), complexity included in the route is simplified sufficiently in the third order, whereas the route shown in Fig. 6(b) includes a fraction of redundancy that is not simplified in the third order MC.

This study deals with a small network which is not complicated at all: 5×5 and 9×9 regular-lattice networks. However, when we consider pickup and delivery VRPs on it, large-size MC networks and small-world properties are embedded in the underlying process. Such virtual or hypothetical complex networks have not been well examined, but beneficial for problems in real societies, like VRPs. Further developments and feasibilities will be discussed in various aspects of network science in the near future.

5 Conclusion

This study is for effective logistic routing design for multiple pickup and delivery systems, associated with RL and artificial potentials as physics-informed functions. Artificial potentials work so well in our reinforcement learning method, and significantly fast solutions are derived in comparison with the classical RL case. Highway-like functions can be alternatively realized by shortcuts, with small-world effects, in which the artificial potential model works successfully. Higher-order MC networks, whose size is so large despite the fact that its lowest-order network is small and simple, are effective for confirmation of route optimization.

Acknowledgements. All the authors thank Air Business Club Co. Ltd. for its support and useful comments, especially by M. Koshiba and T. Oobori, for this study. We also thank the members of Regional ICT Research Center for Human, Industry and Future, The University of Shiga Prefecture, for useful discussion. This work is partially supported by Program to Develop and Promote the Commercialization of Energy Conservation Technologies to Realize a Decarbonized Society from New Energy and Industrial Technology Development Organization and by MEXT/JSPS KAKENHI with Grant Nos. JP22K18704 and 24K21198.

References

1. Fukasawa, R., et al.: Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Math. Program. **06**, 491–511 (2006)
2. Toth, P., Vigo, D.: Models, relaxations and exact approaches for the capacitated vehicle routing problem. Discret. Appl. Math. **123**(1), 487–512 (2002)
3. Pisinger, D., Ropke, S.: A general heuristic for vehicle routing problems. Comput. Oper. Res. **34**(8), 2403–2435 (2007)
4. Li, J., et al.: Deep reinforcement learning for solving the heterogeneous capacitated vehicle routing problem. IEEE Trans. on Cybern. **52**(12), 13572–13585 (2022)
5. Tamagawa, D., Taniguchi, E., Yamada, T.: Evaluating city logistics measures using a multi-agent model. Proc. Soc. Behav. Sci. **2**(3), 6001–6012 (2010)
6. Wu, Y., Song, W., Cao, Z., Zhang, J., Lim, A.: Learning improvement heuristics for solving routing problems. IEEE Trans. Neural Netw. Learn. Syst. **33**(9), 5057–5069 (2022)
7. Zolfpour-Arokhlo, M., Selamat, A., Hashim, S.Z.M., Afkhami, H.: Modeling of route planning system based on Q value-based dynamic programming with multi-agent reinforcement learning algorithms. Eng. Appl. Artif. Intell. **29**, 163–177 (2014)

8. Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. Manag. Sci. **6**(1), 80–91 (1959)
9. Lin, S., Kernighan, B.W.: An effective heuristic algorithm for the traveling-salesman problem. Oper. Res. **21**, 498–516 (1973)
10. Alharbi, M.G., Stohy, A., Elhenawy, M., Masoud, M., Khalifa, H.A.E.W.: Solving pickup and drop-off problem using hybrid pointer networks with deep reinforcement learning. PLoS ONE **17**(5), e0267199 (2022)
11. Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: a survey. J. Artif. Intell. Res. **4**, 237–285 (1996)
12. Gustafson, N.J., Daw, N.D.: Grid cells, place cells, and geodesic generalization for spatial reinforcement learning. PLoS Comput. Biol. **7**(10), e1002235 (2011)
13. Saar, M., Gilad, T., Kilon-Kallner, T., Rosenfeld, A., Subach, A., Scharf, I.: The interplay between maze complexity, colony size, learning and memory in ants while solving a maze: a test at the colony level. PLoS ONE **12**(8), e0183753 (2017)
14. Arriojas, A., Adamczyk, J., Tiomkin, S., Kulkarni, R.V.: Entropy regularized reinforcement learning using large deviation theory. Phys. Rev Res. **5**(2), 023085 (2023)
15. Ikai, R., Miyagi, S., Sakai, O.: Logistics route planning in agent-based simulation and its optimization represented in higher-order Markov-chain networks. In: Teixeira, A.S., Pacheco, D., Oliveira, M., Barbosa, H., Gonçalves, B., Menezes, R. (eds.) Complex Networks XII, SPCOM, pp. 38–50. Springer Nature Switzerland AG, Cham (2021). https://doi.org/10.1007/978-3-030-81854-8_4
16. Sakai, S., Karasaki, T., Ito, T., Murakami, T., Tanaka, M., Kambara, M., Hirayama, S.: Maze-solving in a plasma system based on functional analogies to reinforcement-learning model. PLoS ONE **19**(4), e0300842 (2024)
17. Khatib, O.: Real-time obstacle avoidance for maniputators and mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 25–28. IEEE, St. Liuis (1985)
18. Khatib, O.: The potential field approach and operational space formulation in robot control. In: Narendra, K.S. (ed.) Adaptive and Learning Systems, pp. 367–377. Springer, Boston (1986). https://doi.org/10.1007/978-1-4757-1895-9_26
19. Barabási, A.-L.: Network Science. Cambridge University Press, Cambridge (2019)
20. Liu, F., Janssens, D., Cui, J.X., Wets, G., Cools, M.: Characterizing activity sequences using profile hidden Markov models. Expert Syst. Appl. **42**, 5705–5722 (2015)
21. Saadi, I., Mustafa, A., Teller, J., Cools, M.: Forecasting travel behavior using Markov chains-based approaches. Transp. Res. C **69**, 402–417 (2016)
22. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, 2nd edn. MIT Press, Cambridge (2018)
23. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley, Hoboken (2006)
24. Watkins, CJCH., Dayan, P.: Q-learning. Mach. Learn. **8**, pp. 279–292 (1992)
25. Yamamoto, K., Miyagi, S., Sakai, O.: Order estimation of Markov-Chain processes in complex mobility network embedded in vehicle traces. In: Benito, R.M., Cherifi, C., Cherifi, H., Moro, E., Rocha, L.M., Sales-Pardo, M. (eds.) COMPLEX NETWORKS 2020. SCI, vol. 944, pp. 231–242. Springer, Cham (2021)

Democracy in the Diaspora: The Role of Politics in Migration Networks

Shayan Daijavad(✉) and Theresa Migler

California Polytechnic State University, San Luis Obispo, USA
sdaijava@calpoly.edu

Abstract. Many different factors can influence a person's decision to migrate to a different country. In order to understand the extent to which a country's political environment influences migration, we study the effect of democracy on global migration by constructing a network of international migration flows in 2010 and 2015 and overlaying the Economist's Democracy Index data onto each country. Our network consists of countries as nodes and migration flows as edges, weighed using both raw number of migrants and percentage of origin population. We find results that imply migrants tend to migrate towards countries with higher democracy indexes, and that countries of lower democracy index tend to push migrants towards countries of higher democracy index.

Keywords: political networks · migration networks · democracy index

1 Introduction

Global human migration can be caused by several factors. They include "push factors", pushing people away from their homeland due to poor conditions there, such as poverty, social unrest, or persecution. They also include "pull factors", pulling in people to a new country due to improved living conditions there, such as higher income, social services like education or health care, or a more stable political environment. We analyze the impact of politics on migration by comparing data on the level of democracy of each country and data of human migration between those countries.

2 Related Work

There is a vast amount of work in the field of migration networks, including the works of Danchev and Porter [7], whose thorough analysis of migration network methodologies inspired this work, using countries as nodes and weighting edges with the raw number of migrants. They discuss diagnostics useful in analyzing the structure of migration networks, including degree distribution, clustering coefficient, and betweenness centrality.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 328–339, 2025.
https://doi.org/10.1007/978-3-031-82439-5_27

Maoz gives insight into the formation of international networks and how different countries interact with each other from an economic and political standpoint.[8] Although this work does not discuss migration networks, it does provide insight into the effect of democracy on networks of this type. They find that if democratic countries are strategically grouped together, they are less likely to involve themselves in international conflict. Since conflict and migration are often intertwined, this understanding of country to country interactions is crucial to our own work.

Abel's work involves estimating migration flows between countries and then applying network science to construct a graph with countries as the nodes and edges weighted as the number of migrants travelling between the countries divided by the origin country population.[4][3][2]. This methodology is the same as our own, and we use his derived estimates for migration flows to construct our own network before doing different analysis of it. In a 2019 paper [4], his group created a global network and then divided it into sub-networks composed of groups of countries using the information theoretic community detection algorithm. In a 2014 paper, Abel conducted similar work along with Nikola Sander, categorizing movements between country groups, finding that flows from Asian and Latin America are more focused than in Europe. Europe has the most diverse flows to and from all other regions, whereas Latin America migrants go to North America and Southern Europe, and South Asia migrants migrate to Western Asia, North America, and Europe. [2] In Africa, they found that the majority of migrants move within Africa itself. They also noted that there are flows that go to non-neighboring regions, moving country populations to areas of higher income.

Our work analyzes the effects of democracy from a network perspective, but similar work has been done from a statistical perspective by Kalam Azad [5]. Azad applies a panel data approach using migrant stock data from 189 countries between 1960 and 2015. His approach uses various statistical models to control directly for the effect of democracy on migration, finding a strong 29% increase in migration due to democracy in the long term. However, this analysis is done on a global scale, and is not done with regard to any specific migration flow between two countries.

Similar approaches to Azad's panel data approach include Tranos, Gheasi, and Nijkamp's [10] analysis of a migration network composed of 32 countries all belonging to the Organisation for Economic Co-operation and Development (OECD), an intergovernmental organization centered around economic progress and world trade. They employ both panel data techniques as well network regression analysis. Their findings support the idea that colonialism and common language tend to have a positive effect on migration flows between countries, and more prosperous countries tend to pull more migrants while countries with pools of young people tend to push more migrants. They also note that higher levels of education can cause both pull and push effects.

Social network approaches include Michael Windzio's work [12] as well as Cassie McMillan's [9] work. Both Windzio and McMillan use exponential random

graph models (ERGMs), a technique that allows for sophisticated analysis of how a network is formed. In Windzio's case, the ERGMs are used to analyze the roles of various factors such as economics and linguistics in the formation of migration between countries for the years 1990–2013. His findings include that factors such as religion, language, and being located nearby have a positive effect on migration. Relevant to our own findings, Windzio also found that economics plays a large role in migration, with migration going from poorer countries to richer ones [12]. While his work does not directly analyze politics, the factors he considers are closely related and equally important in a person's decision to migrate. McMillan applies ERGMs in a similar way, and finds similar results in regards to economic factors, as well as strong migration flows between former colonies and their former colonizer. Most relevant to our findings, McMillan found that countries with the same political regime were more likely to be linked by a strong flow, but was not related to the development of weaker flows [9].

Additional work studying political migration networks includes O.C.S. Wilson's 2013 paper [11], in which democracy index (and other statistics, such as income level) are used to do a cross sectional analysis of migration between Malaysia and Asia Pacific countries. Wilson found that level of democracy had an impact on emigration, but not immigration. Moreover, level of income and real interest rates (two economic measures that are closely tied to a country's politics) had the same effect, impacting emigration but not immigration.

3 Methods

Our network is constructed using Abel and Cohen's migration flow estimates[3]. Their dataset includes data from 1990–2015, with five different methods of estimation. For our analysis, we chose to use the closed demographic accounting, pseudo Bayesian estimate, which Abel and Cohen note as the estimate most correlated with real world data. Their data has notable limitations, such as most countries not having a migration flow estimate due to limited data. As such data is either non-existent or difficult to access, their estimates are a reasonable approximation of global migration.

Using Abel and Cohen's dataset, we constructed a directed graph formally defined as follows:

Let $G = (V, E)$ be a directed graph where:

- V is the set of vertices (nodes), where each vertex $v \in V$ represents a country.
- E is the set of directed edges, where each edge $e = (u, v) \in E$ represents a migration flow from country u to country v.

For an illustration of a subgraph of this network see Fig. 1, we include a clique on vertices representing five European countries, France, Germany, Greece, Italy, and Spain. Here the directed edges are weighted with the raw number of migrants.

We consider networks with edges weighted with both raw number of migrants as given by the dataset, as well as edges weighted with a percentage calculated as

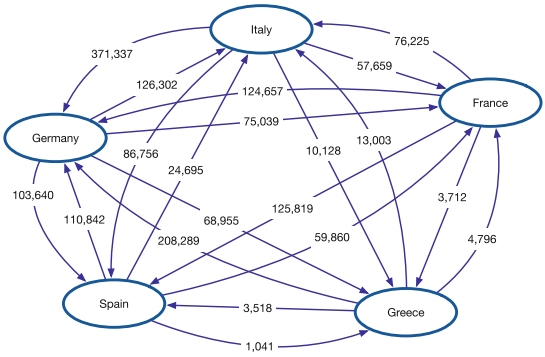


Fig. 1. European Migration Clique (Raw Number of Migrants)

the raw number of migrants divided by the population of the country of origin. Edges are only included if their weight is above 10000 raw migrants or 0.001 percent for percentage of origin population. The population data was sourced from Sourav Banerjee's Kaggle dataset for the years 2020, 2015, 2010, 2000, and 1990 [6].

In order to conduct political analysis, we overlaid the Economist's democracy index values onto each node as an additional attribute to country name. The democracy index is a measure from 1–10 of the level of democracy in over 167 locations, measured by the Economist Intelligence Unit. [1] It is determined by averaging the scores of 60 individual indicators, which can be grouped into five categories: electoral process and pluralism, civil liberties, the functioning of government, political participation, and political culture. The scores for each indicator are determined both through expert opinion as well as public opinion surveys, such as the World Values Survey. After being scored, countries are placed into one of four categories: full democracies, flawed democracies, hybrid regimes, and authoritarian regimes. Full democracies are scores greater than 8 (i.e. New Zealand and Canada), flawed democracies are between 6 and 8 (i.e. Argentina and Botswana), hybrid regimes are between 4 and 6 (i.e. Kenya and Thailand), and authoritarian regimes are less than 4 (i.e. Saudi Arabia and Cuba).

We visualize the percentage of the world population immigrating and emigrating to each democracy index in Fig. 2. While the plot is only for 2015, the plot for 2010 is similar, implying that a large portion of migrants tend to go to countries of high democracy index, but do not leave from countries of any particular index.

We used multiple network related techniques of analysis, including measuring the betweenness centrality, clustering coefficient, and degree distribution of every country. We analyzed all three metrics for both weighted and unweighted edges. We also colored all edges either blue or red for whether or not the edge was going to a country with higher democracy index or lower democracy index (blue for

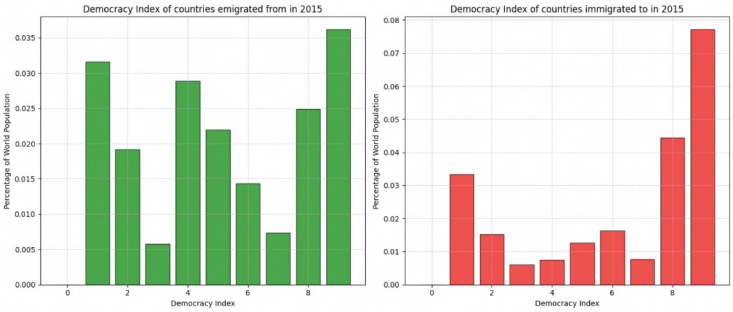


Fig. 2. World Population Percentage vs Democracy Index

higher, red for lower), and then counted the total number of these edges. Degree distribution also includes a count of how many of the edges for each country are to a country of higher democracy index.

See Fig. 3 for the migration network in 2015, with edges weighted with the raw number of migrants flowing between countries. The edges are colored red and blue with the method described above.

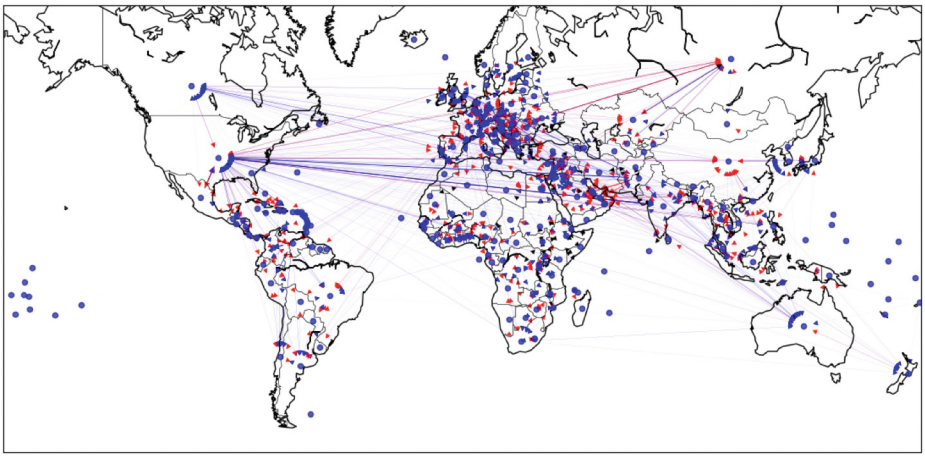


Fig. 3. Migration Network 2015 (Raw Number of Migrants)

In 2015, the total number of nodes (countries with significant migration) was 229, and the total number of edges was 1405 when weighted with raw number of migrants. In 2010, the total number of nodes was also 229, and the total number of edges was 1398.

4 Results

4.1 Degree Distribution

We first present the degree distributions for a global view of the network. We initially consider the unweighted network in determining degree. An unweighted edge exists between two countries if there is migration of over 10,000 people from country A to country B. See Fig. 4 below for the unweighted degree distribution in 2015.

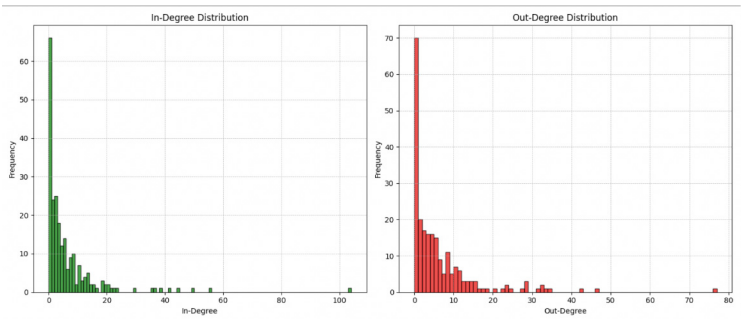


Fig. 4. Unweighted Degree Distribution 2015

We also include the weighted degree distribution for 2015 in Fig. 5, where edges are weighted as raw number of migrants. A weighted edge exists between two countries if the number of migrants is over 10,000. The bins are in increments of 200,000 migrants. The weighted indegree represents the total number of migrants migrating to a country, and and the weighted outdegree represents the total number of migrants leaving a country.

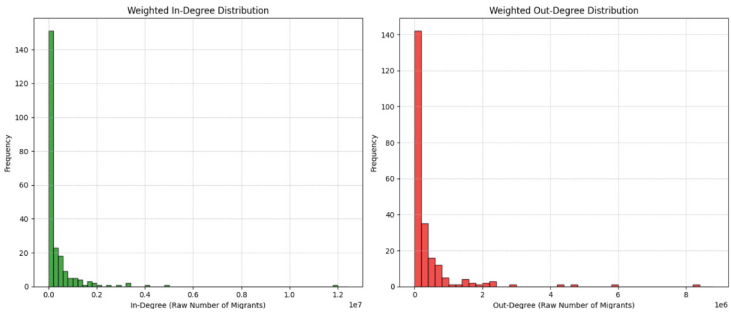


Fig. 5. Weighted Degree Distribution 2015 (Raw Number of Migrants)

Our results for betweenness centrality, clustering coefficient, and degree are summarized in the tables below for the year 2015. The tables show the values for

the 5 countries with the highest values for each measure. For weighted degree, only the outdegree is shown for the percentage based weights, and in degree is shown for the raw number of migrants edges. Summarized takeaways using information from all the data, including 2010 data, are given below each set of tables. Data is N/A if no democracy index is available for the country, and such countries are not included in any of the "from" or "to" columns for degrees, accounting for why those columns do not sum up to the total degree.

4.2 Betweenness Centrality

Betweenness centrality is a measure of how many times a given node appears in the shortest path between two other nodes. In the context of migration, countries with high betweenness centrality are ones that serve as a temporary stopping point for migrants before they continue to another country. They play an important role in the network, acting as a hub between two countries that might otherwise not be directly connected. See Table 1 for the top five countries with respect to betweenness centrality when edges are weighted with the raw number of migrants.

Table 1. Fives Countries with Highest Betweenness Centrality 2015 (Raw Number of Migrants)

Country	Betweenness Centrality	Democracy Index
United States	0.124	8.05
United Kingdom	0.065	8.31
France	0.062	7.92
Italy	0.052	7.98
Canada	0.051	9.08

Table 2 also measures betweenness centrality, but in the network that weighs its edges as the number of migrants divided by the origin country's population.

Table 2. Five Countries with Highest Betweenness Centrality 2015 (Percentage of Origin Population)

Country	Betweenness Centrality	Democracy Index
Spain	0.127	8.3
Syrian Arab Republic	0.117	1.43
Israel	0.106	7.77
Saudi Arabia	0.098	1.93
Iraq	0.096	4.08

With regards to betweenness centrality across both 2010 and 2015, we found that western countries feature prominently when weighing edges with raw number of migrants, whereas more Middle Eastern countries and African countries appear alongside some European countries when weighed with percentage of origin population. This implies that western countries tend to have the largest number of migrants using them as intermediaries for migrating to other countries, whereas Middle Eastern countries and African countries are intermediaries for a large portion of the migrants of individual countries Table 3.

4.3 Clustering Coefficient

Clustering coefficient for a given node is a measure of how connected its neighbors are. In the context of migration, countries with high clustering coefficient values are those who form a group that has more migrants travelling between members of that group than elsewhere. The easier it is to travel between members of a group of neighbor countries, the higher the clustering coefficient is likely to be Table 4.

Table 3. Five Countries with Highest Clustering Coefficient 2015 (Raw Number of Migrants)

Country	Clustering Coefficient	Democracy Index
Guatemala	0.088	5.92
Uzbekistan	0.055	1.95
Georgia	0.048	5.88
Kazakhstan	0.043	3.06
Kyrgyzstan	0.043	5.33

Table 4. Five Countries with Highest Clustering Coefficient 2015 (Percentage of Origin Population)

Country	Clustering Coefficient	Democracy Index
Cook Islands	0.065	N/A
Tokelau	0.036	N/A
Samoa	0.035	N/A
Montserrat	0.034	N/A
Tonga	0.033	N/A

Clustering coefficient using raw number of migrants shows Central Asian countries such as Uzbekistan and Tajikistan featuring prominently, along with American countries such as Guatemala and Mexico. For percentage of population edges, island nations such as the Cook Islands become far more prominent.

4.4 Degree

In what follows, we make a careful distinction between the indegree and outdegree of countries. Note that the indegree of a country is the number of migrants coming into the country, while the outdegree is the number of migrants leaving a country.

In Tables 5 and 6 we inspect countries with highest outdegree in both the raw and percentage graphs. We consider the democracy index of the countries that people are moving to with respect to the origin country.

Table 5. Five Countries with Highest Weighted Outdegree Distribution 2015 (Raw Number of Migrants)

Country	Democracy Index	Weighted Outdegree	To Higher Democracy Index	To Lower Democracy Index
Pakistan	4.4	8,344,677	4,770,898	3,298,753
United States	8.05	5,817,759	747,935	4,917,851
India	7.74	4,717,701	2,506,031	2,211,669
Venezuela	5	4,262,072	4,238,739	0
Germany	8.64	2,955,994	410,926	2,545,068

Table 6. Five Countries with Highest Weighted Outdegree Distribution 2015 (Percentage of Origin Population)

Country	Democracy Index	Weighted Outdegree	To Higher Democracy Index	To Lower Democracy Index
Montserrat	N/A	0.2279	N/A	N/A
Marshall Islands	N/A	0.2020	N/A	N/A
Lebanon	4.86	0.201	0.128	0.073
American Samoa	N/A	0.198	N/A	N/A
United Arab Emirates	2.75	0.185	0.179	0.006

In Tables 7 and 8 we inspect countries with highest indegree in both the raw and percentage graphs. We consider the democracy index of the countries that people are moving from with respect to the destination country.

Table 7. Five Countries with Highest Weighted Indegree Distribution 2015 (Raw Number of Migrants)

Country	Democracy Index	Weighted Indegree	From Lower Democracy Index	From Higher Democracy Index
United States	8.05	11,953,222	10,619,323	747,935
Germany	8.64	4,889,992	4,599,032	410,926
Russia	3.31	4,097,846	1,406,279	1,420,203
United Kingdom	8.31	3,337,302	2,774,019	909,160
India	7.74	3,208,253	2,778,121	2,506,031

Table 8. Five Countries with Highest Weighted Indegree Distribution 2015 (Percentage of Origin Population)

Country	Democracy Index	Weighted Indegree	From Lower Democracy Index	From Higher Democracy Index
United States	8.05	2.012	0.575	0
United Kingdom	8.31	0.614	0.127	0.012
Australia	9.01	0.403	0.111	0.004
New Zealand	9.26	0.390	0.0316	0
Russia	3.31	0.356	0.084	0.007

Key takeaways are that countries with a high democracy index have few edges to countries of higher democracy index. However, countries with low democracy index generally have a majority of their edges going to countries with higher democracy index. Moreover, when the edges are weighted as percentage of origin population, western countries such as the United States or European countries do not tend to have the highest outdegree, and are instead replaced by Middle Eastern countries.

5 Conclusions and Future Work

Our results imply that migrants tended to migrate to countries with higher democracy indices in 2010 and 2015. However, we find that they do not necessarily migrate from any particular democracy index, although countries with high democracy index pull many migrants from countries of lower democracy index, while countries of lower democracy index push many migrants towards countries of higher democracy index. Future work might include identifying what minimum level of democracy is required to reduce the number of migrants being pushed away. There could potentially be a level of democracy that, once met, no longer drives the behavior of migrants. Finding what policies separates that level of democracy from the levels below it would provide insight into what policies keep people satisfied enough to stay within the country. Moreover, we do not divide our own network into communities for our analysis, which could be a potentially valuable way to continue our work, in order to analyze groups of countries with similar levels of democracy and how they interact with each other.

6 Threats to Validity

There are notable limitations to our analysis. First, the dataset we use for migration flows is only an estimate, as real estimates for global migration is often unavailable or not recorded. This could affect the accuracy of the migration network, as it would favor countries with data over those without it. It is possible that less democratic countries are less likely to publish migration data, which would mean that key players might be missing from our network and therefore our political analysis. To mitigate such issues, future research might choose to focus on a smaller region of the world with more complete data for their analysis.

Moreover, while the Economist has given a democracy index to a large portion of the world's countries, it has not done so for most island nations, which tend to be important players in the networks created using percentage based edges.

Our work does not apply any statistical analysis that might be useful in isolating the effect of politics on migration. Instead, we simply overlay the data. There are many other factors that drive migration, and as such our analysis can make no judgement as to the extent to which politics has an individual effect. However, given that a country's politics is often closely tied to a country's economy, as well as other factors that might be considered in migration, our analysis is potentially valuable for future work that ties all factors together.

Although our analysis focuses on the role of democracy in migration, it is important to acknowledge that other factors, such as economic conditions, education, and conflict, are also significant drivers of migration. These factors may correlate with the democracy index, as democracy is often associated with economic development and stability. As a result, our findings may reflect the combined influence of these variables rather than the isolated effect of democracy. Future work should aim to disentangle the effects of democracy from these other factors, for instance, by applying multivariate statistical methods, or by analyzing the effect of a country's change in democracy index over time.

Acknowledgements. This work is a continuation of work done with Sean Du, Kelly Cheung, and Garrett Green, who contributed greatly to the creation of the migration network and the tools used to analyze it.

References

1. Democracy Index 2023 — Economist Intelligence Unit — eiu.com. https://www.eiu.com/n/campaigns/democracy-index-2023/
2. Abel, G., Sander, N.: Quantifying global international migration flows. Science (New York, N.Y.) **343**, 1520–2 (2014). https://doi.org/10.1126/science.1248676
3. Abel, G.J., Cohen, J.E.: Bilateral international migration flow estimates for 200 countries. Sci. Data **6**(1), 82 (2019). https://doi.org/10.1038/s41597-019-0089-3
4. Abel, G.J., DeWaard, J., Ha, J.T., Almquist, Z.W.: The form and evolution of international migration networks, 1990–2015. Popul. Space Place **27**(3), e2432 (2021). https://doi.org/10.1002/psp.2432., https://onlinelibrary.wiley.com/doi/abs/10.1002/psp.2432
5. Azad, K.: The effect of democracy on migration: a panel data approach. MPRA Paper 105058, University Library of Munich, Germany (2020). https://ideas.repec.org/p/pra/mprapa/105058.html
6. Banerjee, S.: World Population Dataset — kaggle.com. https://www.kaggle.com/datasets/iamsouravbanerjee/world-population-dataset/data
7. Danchev, V., Porter, M.A.: Migration networks: applications of network analysis to macroscale migration patterns (2020). https://arxiv.org/abs/2002.10992
8. Maoz, Z.: Networks of nations: the evolution of structure and effects of international networks, pp. 1816–2001 (2009)

9. McMillan, C.: Connecting the dots: using social network analysis to disentangle the factors driving international migration. Int. Migr. https://doi.org/10.1111/imig.13234, https://onlinelibrary.wiley.com/doi/abs/10.1111/imig.13234
10. Tranos, E., Gheasi, M., Nijkamp, P.: International migration: a global complex network. Environ. Plann. B: Plann. Des. **42**, 4–22 (2015). https://doi.org/10.1068/b39042
11. Wilson, O., Thuraisingam, A.S., Baharom, A., Habibullah, M.S.: The impact of democracy on migration. All rights reserved. No part of this publication may be reproduced, distributed, stored in a database or retrieval system, or transmitted, in any form or by any means, electronics, mechanical, graphic, recording or otherwise, without the prior written permission of Universiti Malaysia Sabah, except as permitted by Act 332, Malaysian Copyright Act of 1987. Permission of rights is subjected to royalty or honorarium payment. **7**, 848 (2013)
12. Windzio, M.: The network of global migration 1990–2013: using ergms to test theories of migration between countries. Soc. Netw. **53**, 20–29 (2018). https://doi.org/10.1016/j.socnet.2017.08.006., https://www.sciencedirect.com/science/article/pii/S0378873316300661

Network Embedding

Topological Network Alignment Using Self-supervised Siamese Model

Aljohara Almulhim and Mohammad Al Hasan(✉)

Indiana University, Indianapolis, IN, USA
{aalmulhi,alhasan}@iu.edu

Abstract. Network alignment aims to find the optimal nodes correspondences across two networks leveraging topological and/or attributes information. Existing methods of network alignment belong to two main categories: optimization-based methods and embedding-based methods. Optimization-based methods involve solving eigen-decomposition of a similarity matrix, solving quadratic assignment problem via sub-gradient optimization, or using heuristic-based iterative greedy match. Whereas the embedding-based methods utilize a cost function to learn node representation vectors in a latent space followed by efficient node matching. Optimization based methods are more accurate but they are computationally expensive. Embedding-based methods generally exhibit the opposite trend. In this paper, we propose *SST-Align*, which solves the network alignment task by using a hybrid approach. In the first step, *SST-Align* uses graphlet-count based topology signature of the vertices in an iterative greedy matching method for obtaining an initial alignment. Then in the second step, the initial alignment is used as self-supervised labels for learning node embedding by using a siamese neural network on top of a pair of graph convolutional networks. Extensive experiments conducted on six real-life graph alignment datasets demonstrate that our proposed method outperforms the current state-of-the-art graph alignment methods in terms of node mapping accuracy.

Keywords: Self-Supervised Learning · Graph Alignment · Graphlets Features · Topology-Based Matching

1 Introduction

Network alignment is an important task, with widespread applications in various domains, including bioinformatics [5], security analysis [13], and online marketing [6]. In bioinformatics, network alignment can be used for aligning 3-D structures of different proteins for analyzing their folding patterns. Protein-protein interaction networks of different species can also be aligned to understand and predict the functional activities of one protein using information of activities of a corresponding protein in a different species[10]. In security analysis, network alignment can be used to identify criminal actors by aligning their correspondence across different communication and financial transaction channels. In

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 343–355, 2025.
https://doi.org/10.1007/978-3-031-82439-5_28

online marketing, network alignment can be used for predicting the preference of a user profile by aligning his/her activities on different web platforms, or in different browsers.

Based on the node matching criteria, network alignment can be categorized in two different types: attributed network alignment and plain (topology-only) network alignment. For attributed network alignment, in addition to topology, node attributes (or types) can be factored-in to make the node alignment decision. On the other hand, for plain network alignment, the alignment decision is made based on only the topology of nodes, which makes this a more difficult task. When both the networks to be aligned are identical, topology-only network alignment task is at least as hard as solving a graph isomorphism task, as for two isomporphic graphs, a node permutation matrix exists which defines a one-one-one mapping (alignment) from the vertices of the first network to the vertices of the second network. In this work, we focus on solving topology-only network alignment task, henceforth the discussion in the remaining parts of this paper is generally limited to this variant of network alignment.

Existing solutions of network alignment task can be divided in two main groups: Optimization-based and Embedding- based. An optimization-based method devises a cost function that aggregates the cost of alignment between a pair of nodes, each from one of the two networks and then solves the optimization task using greedy approaches. On the other hand, an embedding-based approach maps the nodes of both the networks in a joint latent space where the aligned node-pairs are positioned in close proximity. For the optimization-based framework, the most fundamental cost function can be the squared Frobenius norm of $A_1 - PA_2P^T$, where A_1 and A_2 represent the adjacency matrices of two networks, and P is the node permutation matrix, which is to be learnt through optimization. However, this framework makes it a combinatorial optimization task, which is often difficult to solve, so most of the optimization-based approaches first compute a node-to-node similarity matrix across the two networks, and then devises a bi-partite matching (aka assignment) based optimization to solve it. Many of these methods suffer from the lack of scalability as the similarity matrix computation task and the parameters of the optimization task grows quadratically with the number of nodes in the network. The solution quality of these methods vary widely depending on the quality of the features used for computing the similarity matrix and also on the quality of the greedy matching criteria that are used for the optimization. On the other hand, the embedding-based methods are generally faster as they use gradient descent based local optimization to make sequential local update to the embedding vectors until convergence. A key challenge for embedding based method is the unavailability of supervised labels, which are needed to design an effective cost function for learning node embedding.

In this work we have developed a hybrid node alignment method, called *SST-Align*, which unifies both optimization and embedding based strategies, in two sequential steps. In the first step, *SST-Align* uses graphlet-count based topology signature of the vertices in an iterative greedy matching method for obtaining

an initial alignment. Then in the second step, the initial alignment is used as self-supervised labels for learning node embedding by using a siamese neural network on top of a pair of graph convolutional networks. Our contributions can be summarized as follows:

- We propose *SST-Align*, a hybrid graph alignment approach utilizing both optimization and embedding approaches. To the best of our knowledge our work is the first which take advantage of both the above paradigms of network alignment in an innovative setting.
- *SST-Align* uses self-supervision to train a novel neural network which uses siamese network on top of twin graph convolution networks.
- Experimental results over six datasets and seven baseline methods show that *SST-Align* yields superior alignment over the state-of-the-art methods.

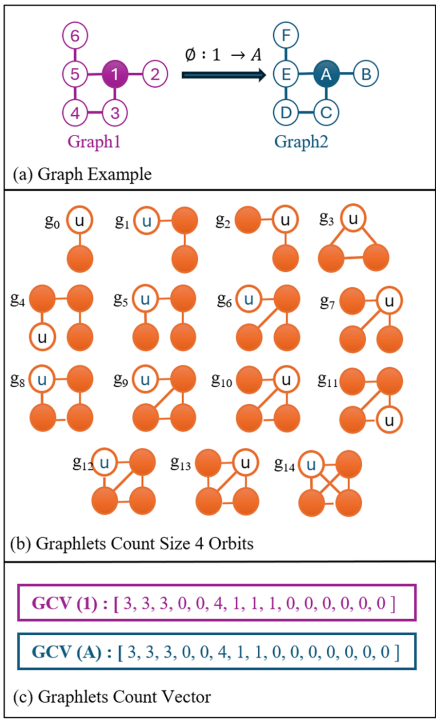


Fig. 1. Illustration of graphlet count vector for graph alignment: (a) Two toy networks Graph1 and Graph2 are shown; Graph1 vertices are labeled with numbers and Graph2 vertices are labeled with letters. For graph alignment, Vertex 1 in Graph1 should be mapped to Vertex A in Graph2; (b) All instances of graphlets with size two, three, and four vertices from vertex u's perspective (c) The extracted graphlet count vectors for node 1 and node A are shown, since both nodes have identical local topology in their respective graphs, these vectors are identical.

2 Problem Definition

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two undirected graphs, where V_1 and V_2 represent the sets of vertices, E_1 and E_2 represent the sets of edges. G_1 is often referred as source graph, and G_2 as target graph. Our goal is to develop a unique mapping function, $\phi : V_1 \rightarrow V_2$, which optimizes the alignment of G_1 and G_2's local structures. The optimization objective specifically focuses on preserving the local structure of each node while maintaining consistency within their neighborhoods. This approach aims to minimize discrepancies in neighborhood connectivity, ensuring that corresponding nodes in both networks exhibit identical local structures and connectivity patterns. The mapping function ϕ must be a bijection, guaranteeing that each node in G_1 is uniquely mapped to a node in G_2 , and vice versa which ensures a direct and complete correspondence between the two graphs, without any overlap or omissions. Thus, the problem centers on addressing topology-based network matching, oblivious of prior knowledge or external attributes.

3 Methods

SST-Align aligns nodes from a pair of networks through two stages: in the first stage it establishes a preliminary alignment (a mapping from the nodes in the first network to the nodes in the second network) by employing an optimization process that utilizes a graphlet signature-based distance matrix. In the second stage, it uses the initial alignment as self-supervision for a supervised neural network model, which learns node embedding of both of the networks. Final alignment is then obtained from the learned embedding vectors of both of the networks after the second stage. The first stage is an optimization based method, and the second stage is an embedding based method, which makes *SST-Align* a hybrid approach, drawing benefits from both of the paradigms.

3.1 Stage 1: Optimization Using Graphlet Count-Based Distance

For topology-based network alignment, graphs do not have vertex and edge attributes; so to obtain topological node similarity across the nodes of G_1 and G_2, we first extract node attributes that capture a node's local structure. Various topology-based signatures for vertices exist, such as degree, clustering coefficient, betweenness centrality, and graphlet count vector. Among these, we chose to use the graphlet count vector local to a given vertex [2]. Graphlets are small (typically upto four or five vertices) induced subgraphs of a graph; based on the number of vertices in a graphlet, there is a fixed number of possible graphlet instances that are local to a given vertex. As shown in Fig. 1(b), there are one size-2 (g_0), three size-3 (g_1 to g_3), and eleven size-4 graphlets (g_4 to g_{14}) from the nude u's perspective. For a given node (say, u) in a graph, we can simply count the number of distinct graphlets of a certain kind that node u participates, and then by accumulating these counts in a vector form we can obtain graphlet count

vector for node u. For example, node 1 in Graph1 in this figure participates in three distinct instances of graphlet g_2: those are (2–1–3, 3–1–5, and 2–1–5); on the other hand, this node does not participate in a single instance of graphlet g_3 (node 1 is not part of any triangle shape); so the count of g_2 and g_3 for node 1 in Graph1 is 2 and 0, respectively. In Fig. 1(c), we show the graphlet count vectors for Node 1 and Node A considering the collection of graphlets g_0 to g_{14} as our graphlet features; since both the nodes have identical topology, they have identical graphlet count vectors (GCV). Note that one can consider graphlet features of larger graphlets to capture more detailed features. If we consider graphlets upto 5 nodes, we can obtain 73 distinct graphlets; but we found that the larger-sized graphlets are not beneficial for our task as many of the graphlet counts are zero; besides, with larger sized graphlet, the local context around a node gets diffused, which diminishes the marginal benefit of adding size-5 graphlets. We used Orca software [9] to obtain the graphlet counts.

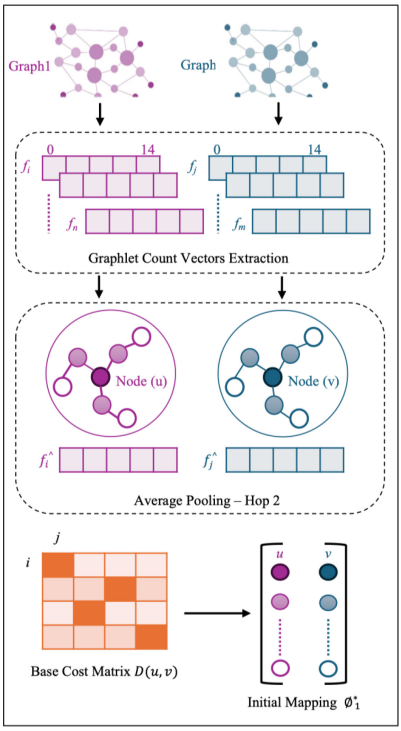


Fig. 2. Overview of the first stage of *SST-Align*: First, graphlets vectors are extracted. Then the average-pooling is applied. Finally, distance matrix is computed from which the source and target nodes are matched using Hungarian Method.

GCV captures the local topology of a node accurately and by matching GCV of the nodes in two different graphs we can obtain a graph alignment solution.

However, in a large graph, similar local topology arises in different regions of the graph, so many nodes in a single graph may have identical GCV, which makes the alignment task difficult. To overcome this issue, we want GCV of a node to be a linear combination of GCV of its k-hop (we considered upto 2nd-hop neighbors) neighboring nodes. This process makes GCV of two nodes in a network different, even if they have identical local topology (considering the graphlet attributes). This is inspired by the concept of graph convolution, which is very popular in graph neural networks [7]. If f_i is graphlet count vector of node u in G_1 and $\mathcal{N}_u^{(k)}$ is the set of neighbors of u up to k hops, the processed GCV, $\hat{\mathbf{f}}_u$ is derived as below: $\hat{\mathbf{f}}_u = \frac{1}{|\mathcal{N}_u^{(k)}|} \sum_{i \in \mathcal{N}_u^{(k)}} f_{u,i}$. After computing GCV of all nodes of both the graphs, we can then compute a distance matrix D by considering the Euclidean distance between GCV of a pair of nodes u and v, where $u \in G_1.V_1$ and $v \in G_2.V_2$; i.e., $D(u, v) = \left\| \hat{\mathbf{f}}_u - \hat{\mathbf{f}}_v \right\|$.

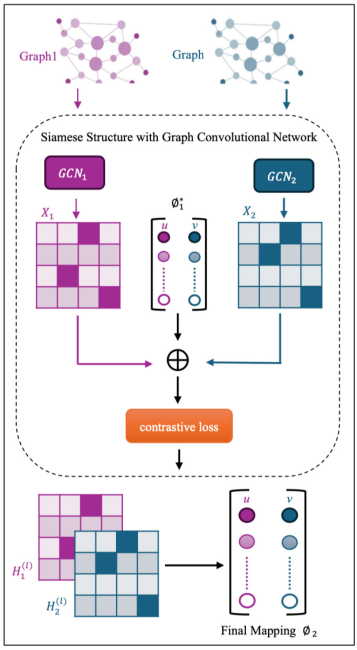


Fig. 3. An illustration of the embedding framework of *SST-Align*. A pair of graph convolutional networks are used in siamese fashion to learn embedding vectors of vertices of both the graph by using a self-supervised learning setup. Finally, embedding vectors of vertices across the graphs are matched by using kd-tree, which aligns each node from Graph1 to exactly one node in Graph2, which has the most similar embedding.

If we consider the distance matrix as a cost matrix, the graph alignment problem then reduces to an assignment (aka, bi-partite matching) task, a classical combinatorial optimization problem. The overall objective is to match each of

the nodes of the first graph to a distinct node in the second graph so that the aggregated cost of the matched node-pair is minimized. Say, ϕ_1^* is the optimal assignment; then assignment task is to solve the following optimization problem:

$$\phi_1^* = \arg\min_{\phi_1} \sum_{u \in V_1} D(u, \phi_1(u)) \tag{1}$$

To solve this we use well-known Hungarian method [12], which yields a global optimal solution. This completes the first stage of *SST-Align*. The overall tasks in the first stage is shown in Fig. 2.

A key computational challenge in this stage is to compute the graphlet-based distance matrix. Since the matrix have distance values for all pair of vertices, even for moderate-sized graphs, the number of entries in the matrix exceeds millions, and the cost grow quadratically with the number of vertices in the graphs. To overcome this challenge, we do not compute the full distance matrix for a node u in the source graph, we rather find p (a user-defined parameter) fraction of most similar (least distance) nodes in the target graph (G_2) and compute those distances only. For instance if $p = 0.25$, only 6.25% entries in the distance matrix is computed. To accomplish this we utilize Kd-tree, a space-partitioning data structure for organizing points in a k-dimensional space. By indexing the vertices in the target graph in the KD-tree, we run a distinct nearest neighbor query for every nodes in the source graph to retrieve p fraction of the vertices in the target graph that are nearest to the given query node. The unfilled entries of the distance matrix is populated with large value, or ignored if the optimization library for Hungarian algorithm allows sparse matrix format input for the cost matrix. This simple trick alleviates the computational challenge issue substantially. We also use Kd-tree based search in the second stage of *SST-Align*, when we obtain the final alignment between the source and destination graph.

The outcome of this stage is ϕ_1^*, which provides a preliminary alignment of nodes from G_1 to G_2. Although this mapping is optimal for the cost matrix, it is not perfect for graph alignment as graphlet-feature based distance is imperfect. But, this mapping can serve as guidance for learning in a subsequent step. We consider the generated preliminary mapping as pseudo labels that provide training signals for supervised embedding of vertices in a latent space for enhancing alignment accuracy. Since this supervised labels are generated automatically without human intervention, we call the subsequent embedding model a self-supervised model. We discuss this in the following sub-section.

3.2 Stage 2: Learning Representation of Nodes of Both Graphs Using Convolutional Neural Network Within Siamese Structure

In this stage we use graph convolution network (GCN) to embed the vertices of both the graphs in a unified latent space with the expectation that a node u in the source graph, and its desired aligned node v from the target graph will be embedded in very close proximity in the latent space. In our model, we

place two GCNs side-by-side using a Siamese network structure as shown in Fig. 3. The justification of this setup is to utilize individual GCNs to embed the vertices of each of the graphs by considering locality, and then use the Siamese setup to capture node similarity across the two networks through embedding vectors. We use a classification-based loss function and use the initial mapping ϕ_1^* derived from the previous phase to serve as labels (y) for the purpose of training. In addition to the graph adjacency lists, GCN also requires node features (X_1 for the source graph, and X_2 for the destination graph). We use graphlet feature vector (computed in first stage) as node features. Two GCN models are trained in parallel within a Siamese network framework, adhering to the standard layer-wise propagation rule of GCN. The propagation rules for each model are given by the equations below:

$$H_1^{(l+1)} = \sigma\left(\hat{D}_1^{-\frac{1}{2}}\hat{A}_1\hat{D}_1^{-\frac{1}{2}}H_1^{(l)}W_1^{(l)}\right) \tag{2a}$$

$$H_2^{(l+1)} = \sigma\left(\hat{D}_2^{-\frac{1}{2}}\hat{A}_2\hat{D}_2^{-\frac{1}{2}}H_2^{(l)}W_2^{(l)}\right) \tag{2b}$$

In this formulation, $H_1^{(l+1)}$ and $H_2^{(l+1)}$ denote the features at layer $l+1$ for the first and second models, respectively. The feature matrices $H_1^{(0)} = X_1$ and $H_2^{(0)} = X_2$ are used as the initial features for each model. The activation function is represented by σ, while the matrices $\{\hat{A}_1, \hat{A}_2\}$ and $\{\hat{D}_1, \hat{D}_2\}$ correspond to the adjacency matrix and the degree matrix. The trainable weight matrices for each layer, $W_1^{(l)}$ and $W_2^{(l)}$, are tailored to each model. The loss function employed is a contrastive loss, as depicted in Eq. 3, which binds the output embedding vectors of source and target nodes derived from the GCN models:

$$\begin{aligned} L(u, v, y_{uv}) = \; & y_{uv} \cdot \frac{1}{2}\left(\|z_u - z_v\|^2\right) \\ & + (1 - y_{uv}) \cdot \frac{1}{2}\left(\max(0, m - \|z_u - z_v\|)\right)^2 \end{aligned} \tag{3}$$

where z_u and z_v are the embedding vectors for nodes u in G_1 and v in G_2. The binary variable y_{uv} indicates the relationship between these nodes; $y_{uv} = 1$ shows that u and v are linked, while $y_{uv} = 0$ indicates no connection. The margin parameter m serves as a threshold to establish the minimum distance between the embedding of nodes that are not connected. This loss function is designed to ensure that nodes without connections are spaced sufficiently apart, while similar nodes are drawn closer together in the embedding space. To optimize the embedding and the associated parameters within our model, we define the following optimization objective:

$$\Theta^* = \arg\min_{\Theta} \sum_{(u \in G_1, v \in G_2, \phi_1^*) \in \mathcal{D}} L(u, v, y_{uv}; \Theta) \tag{4}$$

Through this process, the model learns to generate embedding vectors that effectively reflect the connectivity patterns within and across the graph data.

Finally, we obtain our final alignment ϕ_2 using k-d tree matching as shown below:

$$\begin{aligned} H_1^{(l)} = GCN1(G_1), \quad H_2^{(l)} = GCN2(G_2) \\ \phi_2 = \text{k-d tree matching}(H_1^{(l)}, H_2^{(l)}) \end{aligned} \tag{5}$$

In Algorithm 1, we present the pseudo-code, which illustrates the precise steps of *SST-Align*. The source code and datasets are available in the public repository at https://github.com/aalmulhi/SST-Algin.

Algorithm 1. *SST-Align* Algorithm

Require: Two undirected plain graphs: Graph1 $G_1 = (V_1, E_1)$, Graph2 $G_2 = (V_2, E_2)$, p defines the percentage of nodes from V_2 during k-d Tree matching
Ensure: Mapping function $\phi : \mathcal{V}_1 \rightarrow \mathcal{V}_2$

procedure FIRSTSTAGEMAPPING(G_1, G_2, p)
  Extract graphlet count vectors $\{f_u\}$ and $\{f_v\}$ where $u \in V_1$, $v \in V_2$
  $\hat{f}_u = \text{AvgPool}(f_u), \forall u \in V_1$, and $\hat{f}_v = \text{AvgPool}(f_v), \forall v \in V_2$
  $D(u, v) =$k-dTreeMatching$\Big(\{\hat{f}_u\}, \{\hat{f}_v\}, p\Big)$
  $\phi_1^* = \arg\min_{\phi_1} \sum_{u \in V_1} D(u, \phi_1(u))$ by applying Hungarian method on D
end procedure
procedure LEARNING-REPRESENTATION(G_1, G_2, X_1, X_2, ϕ_1^*)
  $y = \phi_1^*$
  $H_1^{(0)} = X_1$, $H_2^{(0)} = X_2$
  $\Theta^* = \arg\min_{\Theta} \sum_{(G_1, G_2, y) \in \mathcal{D}} L(G_1, G_2, y; \Theta)$
  Compute L as contrastive loss
  $H_1^{(l)} = \text{GCN1}(G_1)$, $H_2^{(l)} = \text{GCN2}(G_2)$
  $\phi_2 = \text{k-dTreeMatching}(H_1^{(l)}, H_2^{(l)}, p)$
end procedure

4 Experiments and Results

Datasets. We use six real-world datasets to assess *SST-Align*'s performance in aligning two isomorphic graphs. The datasets are collected from KONECT [1], also known as The Koblenz Network Collection. The platform offers a large collection of network datasets that can be utilized for academic research and network analysis studies. A graph is initially retrieved, and a duplicated graph is constructed by randomly permuting the vertex-id of the nodes in the duplicate graph. We maintain fully connected undirected graph by considering the largest connected component. Table 1 provides a summary of the datasets used in our experiments, including node and edge counts. More details about each dataset are given below: *Moreno Protein-Protein* is a protein-protein network in which nodes refer to proteins in Yeast and links refer to their interactions. *Mayan-Vidal* is a comprehensive map of all the protein-protein interactions in the human proteome. In this map, proteins are represented as nodes and the

interactions between them are shown as connections. *Autonomous Systems (AS-Com)* is a network of autonomous systems, where each autonomous system acts as a node and their connectivity is established by links. *Ca-AstroPh* is scientific collaboration network where authors are represented as nodes and collaborations are represented as links. *Cora* consists of academic publication as nodes, with the citations between them serving as the links.

Table 1. Dataset Summary

Dataset	# Nodes	# Edges	Description
Moreno Propro	1,458	1,948	Proteins Interactions
Maayan Vidal	2,783	6,007	Proteins Interactions
AS Com	6,474	12,572	Infrastructure Network
PGP	10,680	24,316	Contact Interaction
Ca AstroPh	17,903	196,972	Citation Network
Cora	23,166	89,157	Citation Network

Baseline Methods. We evaluate our results by comparing against the following graph alignment models: (1) **Graphlet-Align**[3] is an optimization approach, which utilizes a bi-partite matching algorithm in conjunction with graphlet signatures to acquire knowledge of high-order proximity; (2) **REGAL**[8] employs node representation learning to align nodes between different graphs, after employing the xNetMF embedding method; (3) **GRAAL** [11] captures local graph structures and aligns nodes based on these graphlet signature similarity; (4) **IsoRank**[14] solves the alignment as eigenvelaue problem with relaxed constraints to computes nodes' similarity score; (5) **FINAL**[16] is considered as fast approximation of IsoRank method, which solves the alignment task by leveraging the attributes information into the topology structure; (6) **CONE-Align**[4] develops learning representation approach by combining GCN and Wasserstein Procrustes method; (7) **BRIGHT**[15] integrates random walk with restart and shared linear layers to align nodes across networks with the help of a percentage of anchor nodes.

Experiment Setup. For training GCNs, we employ various batch sizes ranging from 32 to 64 in our experimental setup and trained our models for a duration of 20 to 50 epochs to achieve reliable learning results. To achieve a compromise between convergence speed and training stability, the learning rate was set to 0.0001. To assess the performance of the supervised model, we implement an 80/20 train-validation split. Positive label set is collected from the first stage (which makes the model self-supervised) whereas negative labels are generated by selecting random edges form the target graph. For every positive label, 20 negative ones are generated. Once final embedding is obtained, we utilize Kd-tree data structure over the node embedding vectors to find matched nodes. Our embedding dimension is 128. For IsoRank and Final algorithms, which require

attributes or/and similarity matrix as prior, we extracted nodes degree as features and compute degree-based similarity matrix as prior knowledge for a fair comparison. Moreover, we set *alpha* in these methodologies to be 0.5 where importance of node similarity and network topology are equally treated. Where in BRIGHT, the number of anchor nodes is set to be 5%. We run REGAL with structure-only version. Finally, Graphlet-Align and GRAAL are executed using edge list information.

Metric. We evaluate the performance of our model using Hard Assignment Accuracy (HAA). Each node from the source graph is mapped exclusively to one node in the target graph; then HAA is computed as the ratio of number of correctly mapped nodes and the total node count.

Table 2. Quantitative Results of Alignment Accuracy using HAA. The asterick indicates that results are not reported for methods running beyond a week

Method	Moreno Propro	Maayan Vidal	AS Com	PGP	Ca AstroPh	Cora
SSTAlign	**0.7318**	**0.8272**	0.5420	0.8040	**0.9848**	**0.9521**
Graphlet	0.6914	0.8101	0.5380	0.7272	0.8201	0.9435
REGAL	0.5789	0.7661	0.5176	0.5875	0.8131	0.8970
GRAAL	0.4643	0.8228	0.3577	*	*	*
IsoRank	0.6351	0.8063	0.5378	0.6843	0.7948	0.9361
FINAL	0.7037	0.8193	0.5377	0.7404	0.8233	0.9379
CONEA	0.5247	0.5742	0.5200	0.2388	0.2196	0.4343
BRIGHT	0.7236	0.8265	**0.5600**	**0.8412**	0.8310	0.9426

Comparison Results. Table 2 shows the HAA values of *SST-Align*, along with other competitors for all the six networks that we experiment with. In this table, the columns represent different datasets, while the rows list the methods being compared. Out of six datasets, *SST-Align* performs the best in four datasets, often with good margin. For instance, in Ca-AstroPh dataset, *SST-Align*'s HAA is 0.9848, where the second best method (BRIGHT) achieves 0.8410 HAA value. In fact, in Ca-AstroPh and Cora datasets *SST-Align* shows outstanding HAA values, 0.9848 and 0.9521, respectively. These datasets are the largest among the experimental datasets. This emphasizes that strength of *SST-Align* to align large-scale networks, while depending solely on topology information.

In our experiment, BRIGHT appears to be the second best method with rank 1 in two datasets, and rank 2 in remaining four datasets. It is important to note that although BRIGHT surpasses *SST-Align* in As-Com and PGP datasets, this superiority may be due to its utilization of anchor nodes for node mapping. GRAAL could not complete its computation within 1 week of running time on three of the six datasets.

Ablation Study. To show the effectiveness of different stages of *SST-Align*, we report results of mapping accuracy after stage 1 and after stage 2 in Table 3. As can be seen, all datasets have better performance after Stage 2 than stage 1, improving the accuracy by 1.98% to 19.62%. This validates the utility of an hybrid approach, combining both optimization and embedding based approach.

Table 3. Results of Ablation Study on *SST-Align*Accuracy Comparison

Method	Moreno Propro	Maayan Vidal	AS Com	PGP	Ca AstroPh	Cora
Stage1	0.7065	0.8222	0.5376	0.7451	0.8233	0.9412
Stage2	**0.7318**	**0.8272**	**0.5420**	**0.8040**	**0.9848**	**0.9521**

5 Conclusion

We propose *SST-Align*, a graph alignment method, which achieves high one-to-one mapping accuracy for two provided networks using solely the network topology. *SST-Align* utilizes a hybrid approach combining both optimization and embedding based ideas; It first solves an assignment based optimization over graphlet distance based cost matrix, and then uses a Siamese neural network architecture with twin graph convolutional network to find final alignment. Experimental results show *SST-Align*'s superiority over state-of-the-art methods.

References

1. KONECT: The Koblenz network collection (2023). http://konect.cc/. Accessed 10 Sept 2023
2. Ahmed, N.K., Neville, J., Rossi, R.A., Duffield, N.G., Willke, T.L.: Graphlet decomposition: framework, algorithms, and applications. Knowl. Inf. Syst. **50**, 689–722 (2017)
3. Almulhim, A., Dave, V.S., Hasan, M.A.: Network alignment using graphlet signature and high order proximity. In: Nicosia, G., Pardalos, P., Umeton, R., Giuffrida, G., Sciacca, V. (eds.) LOD 2019. LNCS, vol. 11943, pp. 130–142. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-37599-7_12
4. Chen, X., Heimann, M., Vahedian, F., Koutra, D.: Cone-align: consistent network alignment with proximity-preserving node embedding. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 1985–1988 (2020)
5. Faisal, F.E., Meng, L., Crawford, J., Milenković, T.: The post-genomic era of biological network alignment. EURASIP J. Bioinf. Syst. Biol. **2015**(1), 1–19 (2015). https://doi.org/10.1186/s13637-015-0022-9

6. Geng, Q., Deng, S., Jia, D., Jin, J.: Cross-domain ontology construction and alignment from online customer product reviews. Inf. Sci. **531**, 47–67 (2020). https://doi.org/10.1016/j.ins.2020.03.058., https://www.sciencedirect.com/science/article/pii/S0020025520302383
7. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
8. Heimann, M., Shen, H., Safavi, T., Koutra, D.: Regal: Representation learning-based graph alignment. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 117–126 (2018)
9. Hočevar, T., Demšar, J.: A combinatorial approach to graphlet counting. Bioinformatics **30**(4), 559–565 (2014)
10. Kuchaiev, O., Milenkovic, T., Memisevic, V., Hayes, W., Przulj, N.: Topological network alignment uncovers biological function and phylogeny. J. R. Soc. Interface **7**(50), 1341–54 (2010)
11. Kuchaiev, O., Pržulj, N.: Integrative network alignment reveals large regions of global network similarity in yeast and human. Bioinformatics **27**(10), 1390–1396 (2011)
12. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res. Logist. Q. **2**(1–2), 83–97 (1955)
13. Qin, Y., Xiao, Y., Liao, X.: Vulnerability intelligence alignment via masked graph attention networks. In: Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, pp. 2202–2216 (2023). https://doi.org/10.1145/3576915.3616686
14. Singh, R., Xu, J., Berger, B.: Global alignment of multiple protein interaction networks. In: Biocomputing 2008, pp. 303–314. World Scientific (2008)
15. Yan, Y., Zhang, S., Tong, H.: Bright: A bridging algorithm for network alignment. In: Proceedings of the Web Conference 2021, pp. 3907–3917 (2021)
16. Zhang, S., Tong, H.: Final: fast attributed network alignment. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1345–1354 (2016)

Critical Synthetic Benchmarking of Graph Representation Learning

Noé Durandard[1], Camille Roth[2](✉), and Telmo Menezes[2]

[1] LATTICE, ENS-PSL/CNRS/USN, Paris, France
noe.durandard@psl.eu
[2] CAMS, CNRS/EHESS, Paris, France
camille.roth@cnrs.fr, telmo@menezes.net

Abstract. Graph Representation Learning (GRL) has emerged as a thriving field within machine learning, leveraging Graph Neural Networks (GNNs) to extract meaningful representations of graphs. The typical evaluation framework however relies heavily on a limited set of empirical datasets which appear, each and as a whole, to cover a rather narrow segment of the potential graph space. We propose a novel synthetic benchmarking framework capable of spanning extensive portions of the space of possible graphs, both in terms of basic size and sophisticated topological properties. This helps us demonstrate that several state-of-the-art models exhibit task-specific performance and may not generalize well. For some regions of the space, naive graph representation methods such as degree distributions may perform even better. Our contribution makes it possible to discern regions where computationally intensive GRL methods offer substantial value and, more broadly, emphasizes the need for a broad, tunable benchmark covering an array of graph properties. We encourage the integration of such synthetic benchmarks to complement the empirical appraisal of GRL models.

Keywords: Graph Representation Learning · Graph Neural Networks · Synthetic Benchmarking · Graph Property Embeddings

1 Motivation

Graph Representation Learning (GRL) generally denotes the extraction of graphs features to obtain meaningful vectorized representations of entire graphs, notably through the resurgence of deep learning applied to graphs, or Graph Neural Networks (GNNs). This problem has risen to be a "hot topic" in the Machine Learning community in just a half-dozen years, becoming an extremely fast-growing field with hundreds of papers published yearly [8,21].

The myriad of applications, covering a wide range of domains from quantum mechanics to medicine or social sciences, and their uncontested industrial value, largely drive GNNs' empirical and practical developments, despite a limited theoretical understanding. Admittedly, researchers of the field have engaged

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 356–368, 2025.
https://doi.org/10.1007/978-3-031-82439-5_29

in a race focused on performance improvements. A multitude of papers introducing new models with varying architecture or training strategies are published on a regular basis. The capabilities of the models are usually benchmarked on a reduced set of downstream applications and, accordingly, empirical graphs. Models are generally gauged through graph classification tasks and, in practice, are usually evaluated on a selection of five to ten graph classification datasets, some being very recurrent in the literature. This is highly beneficial to experimental comparability and to keeping the development in this hectic field approximately tractable. Yet, such evaluation schemes also bear limitations. For one, it is indeed legitimate to fear that overbids on a few downstream tasks would contribute to develop models specifically fitted for these tasks, at the expense of more generalizable methods. Moreover, it can be a priori hard to interpret the strengths and weaknesses of different models solely based on few evaluation datasets, making it difficult to choose a model adapted for a target application which either diverges from the said classification tasks or from the usual network topologies that these recurrent datasets feature.

Given the extreme growth of the field, we argue that it is presently desirable to design a broad, tunable and flexible benchmark composed of graphs tightly covering a wide range of fundamental network properties within the space of possible empirical graphs—also noting, as we shall see, that the set of existing benchmark graphs actually occupies a quite specific region of that space.

A relatively naive yet straightforward approach that we explored here consists in representing a diverse spectrum of graphs on a 2D-space based on their number of nodes and edges, and then to use a known generative process to create datasets of graphs with easily interpretable properties.

A review of about a hundred of articles introducing GRL models for whole-graph related tasks allows to gain insights on the current state of the evaluation processes. Then, thanks to existing network generation algorithm, we propose a new customizable and controlled benchmark of synthetic graphs. Performances of two self-supervised learning GNNs are then evaluated on this dataset, along with a very simple baseline based on graphs' degree distribution which turns out to be markedly efficient for a variety of situations.

2 Literature Review

We first perform a review of a significant number of recent papers to examine the diversity or uniformity of the datasets used to evaluate the proposed GRL methods. For this purpose, we manually gathered papers tackling graph-related tasks and focusing on developing methods for graph classification that were up to 5 years old at the initial time of initial data collection (in August 2023). We used a combination of keyword-based Google Scholar queries and further examination of the recurrently-cited models against which results are often being compared. This process resulted in a pool of 105 papers introducing 108 distinct models which, we contend, represent a fair overview of the core standards of the field after a significant period of growth. This rise is reflected by the distribution

of the models' release dates within the pool. Specifically, it includes 1 model published in 2018, 8 in 2019, 11 in 2020, 19 in 2021, and 40 in 2022, with the remaining 29 models released between January and August 2023.

The analysis of the datasets used to gauge the models sheds light on the general evaluation procedure of the developed models. Each paper indeed relies on a number of what we will here uniformly call "*datasets*"—each *dataset* being a collection of graphs which all belong to a similar empirical domain, mainly stemming from real-world data, and which have been manually labeled to differentiate various sub-classes of graphs. GRL models are in turn benchmarked via their ability to properly discriminate graph sub-classes within each dataset. For instance, `PROTEINS` [3] is expectedly a collection of protein networks i.e., networks of amino acids that are linked if they are close in space, associated with binary labels segregating enzymes from non-enzymes; `IMDB-MULTI` [27] is a collection of ego-centered networks of movie actors extracted from the online platform IMDb, where neighbors are linked if they appeared in the same movie, and labels represent the cinematic genre of the movies that were considered to build each actor-centric network; `HIV` [20,25] originates from medical chemistry experiments which enabled to test and document the ability of over 40,000 chemical compounds to inhibit HIV replication or not.

In total, we found 74 distinct datasets across all reviewed papers. Yet, a majority of the datasets are used only in a few papers, 31 listed datasets appear once and only 24 datasets are present in more than five articles. Conversely, the most popular datasets are almost ubiquitous, they are used in a great majority of the papers. For instance, `PROTEINS`, the most popular dataset, is used to evaluate nearly 80% of the reviewed models. Figure 1 illustrates this wide disparity.

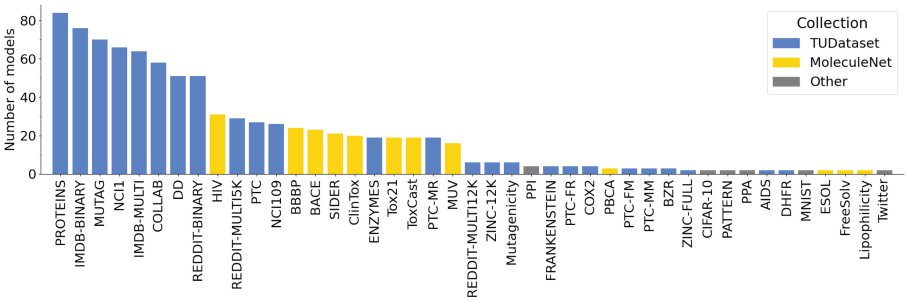


Fig. 1. Number of occurrences of each dataset within the studied pool of papers, considering only datasets with at least two appearances. The membership of a dataset to one of the two mainstream dataset collections is figured by the bar color.

Broadly, three classes of popularity emerge from this analysis. The first 8 datasets are prevalent, appearing in almost half of the reviewed papers or more (`REDDIT-BINARY`, `DD`, `COLLAB`, `IMDB-MULTI`, `NCI1`, `MUTAG`, `IMDB-BINARY`, `MUTAG` from the least to most popular). Then, 13 other benchmarks are commonly

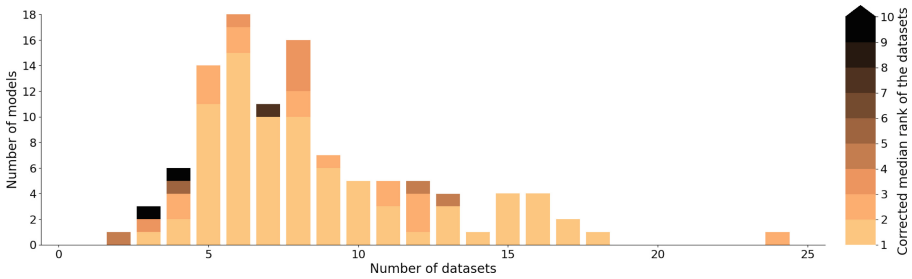


Fig. 2. Distribution of the number of datasets used per paper of our pool. Each bar denotes the number of models whose evaluation relies on a certain number of datasets, while the color of each block within that bar is associated with the median rank of the datasets used (divided by the minimal median rank possible for a given number of datasets), further illustrating how most papers rely on the most popular datasets.

adopted, with datasets appearing in 30% (`HIV`) to 15% (`MUV`) of the paper pool. Finally, the remaining 53 datasets are more rarely used, with at most 6 occurrences ($\leq 6\%$).

These datasets originate from a multitude of efforts of data collection but are now gathered in collections that strive for accessibility. These collections of datasets i.e., collections of collections of graphs, are often included in libraries that make the datasets easily loadable and practical to integrate within different pipelines which, in turn, normalize GRL validation endeavors. Two collections are particularly commonplace: almost every dataset can be found either in the TUDataset or MoleculeNet collections. The 8 most prevalent datasets may be found in the TUDataset collection, while the next most common set is comprised of both MoleculeNet and TUDataset benchmarks. TUDataset gathers datasets of graphs with diverse nature [17], from small molecules (e.g. `MUTAG`, `NCI1`, `PTC`) and bioinformatics (e.g. `PROTEINS`, `DD`, `ENZYMES`) to social networks (e.g. `IMDB-BINARY`, `COLLAB`, `REDDIT-BINARY`). Then MoleculeNet presents a variety of graph collections stemming from biophysics (e.g. `HIV`, `BACE`, `MUV`), physiology (e.g. `BBBP`, `SIDER`, `ClinTox`) or physical chemistry (e.g. `ESOL`, `FreeSolv`, `Lipophilicity`). In practice, datasets from this collection are often used to inquire transfer learning capabilities of the developed models (e.g., [7,9,22,23,28,30]). On the whole, the existence of these collections contributes to the concentration of the models' evaluation benchmarks, as further illustrated by Fig. 2 which demonstrates the almost uniform focus of most papers on the most popular datasets.

3 A Thick Artificial Benchmark

3.1 Network Coverage of Literature Datasets

While apparently diverse, the datasets against which GNNs are usually benchmarked likely represent a relatively narrow range within the realm of empirical

networks. For one, many of them are rather small-sized, comprising graphs made of markedly less than a hundred nodes; for instance, only three out of the 21 most popular datasets exhibit an average graph size above this threshold (see basic statistics for these datasets in Table 1). Furthermore, while each dataset comprises, by construction, several subclasses of graphs, each typically corresponds to one specific class of networks which all stem from one specific context—proteins, movies, collaborations, etc. This makes them prone to possess similar generative rules and exhibit similar topological features, as demonstrated by a rich literature aimed at classifying graphs [5,16,18,29]. As a result, each dataset is likely to cover a small region of the whole space of possible networks, both in terms of size and in terms of structural properties. Figure 3 provides a simple illustration of this phenomenon by positioning the 8 most used benchmarks on a naive phase space (n, m) featuring their numbers of nodes n and their numbers of edges m. Each of the contours spreads over a limited portion of that space, knowing that each covers one network class only. Put simply, the evaluation region of models is generally narrowly bounded, which is compounded by the fact that most papers heavily rely on a narrow set of datasets among the most used ones: corresponding models are thus evaluated on an even more limited set of network topologies.

Finally, for a given dataset, it is likely that basic network size already provides some classification cues. To the best of our knowledge, very little research has been done specifically in this regard: to which extent does basic network size information i.e., just (n, m) values, help partition the space of graphs within a given dataset? As we shall show below and as is summarized in Table 2, this information actually appears to be surprisingly valuable: solely using (n, m) pairs reaches on average more than 90% than the F1-scores exhibited by GraphMAE or PGCL (resp. 91.5% and 90.7%) on the 8 most popular datasets; similarly, computing the graph degrees' histogram brings this value to more than 95% (resp. 95.9% and 95.3%). It is hence sensible to question whether current datasets make it relatively easy for GNNs to tackle the classification tasks they are presented with. This leads us to propose an orthogonal approach to benchmarking by appraising classification capabilities, on the one hand, *within* very narrow regions of the network size space (instead of using datasets spread on broader regions) and, on the other hand, for *more diverse* graph topologies stemming from a variety of generative principles (instead of just one). The rationale is that an efficient GRL model should first and foremost be able to differentiate similar-sized graphs from distinct realms, or at least with distinct topological features.

3.2 Synthetic Graphs and Size Grid

To this end, we argue that it would be most fruitful to systematically evaluate GNNs against what we denote as a *thick artificial benchmark* (TAB) . This benchmark aims at covering a substantial range of network sizes and densities, while featuring a variety of network generation principles for one given network

Table 1. Summary statistics for the 21 most used datasets. Datasets with average graph size above 100 are emphasized

Dataset Name	# graphs	$\langle n \rangle$	$(\sigma(n))$	$\langle m \rangle$	$(\sigma(m))$	# classes
PROTEINS	1113	39.1	(45.8)	72.8	(84.6)	2
IMDB-BINARY	1000	19.8	(10.1)	96.6	(105.6)	2
MUTAG	188	17.9	(4.6)	19.8	(5.7)	2
NCI1	4110	29.9	(13.6)	32.3	(14.9)	2
IMDB-MULTI	1500	13.0	(8.5)	65.9	(110.8)	3
COLLAB	5000	74.5	(62.3)	2457.8	(6438.9)	2
DD	*1178*	*284.3*	(272.0)	*715.7*	(693.9)	*2*
REDDIT-BINARY	*2000*	*429.6*	(554.1)	*497.8*	(623.0)	*2*
HIV	41127	25.5	(12.1)	54.9	(26.4)	2
REDDIT-MULTI5K	*4999*	*508.5*	(452.6)	*594.9*	(566.8)	*5*
PTC	349	14.1	(8.4)	14.5	(9.4)	2
NCI109	4127	29.7	(13.6)	32.1	(15.0)	2
BBBP	2050	23.9	(10.7)	51.6	(23.7)	2
BACE	1513	34.1	(8.5)	73.7	(17.5)	2
SIDER	1427	33.6	(47.9)	70.7	(98.4)	2
ClinTox	1484	26.1	(15.7)	55.5	(33.8)	2
ENZYMES	600	32.6	(15.3)	62.1	(25.5)	6
Tox21	7831	18.6	(11.3)	38.6	(25.0)	2
ToxCast	8597	18.7	(11.6)	38.4	(25.3)	2
PTC-MR	344	14.3	(9.0)	14.7	(10.1)	2
MUV	93087	24.2	(5.1)	52.6	(11.5)	2

size pair (n, m), hence increasing the resolution and difficulty of the task compared with standard datasets. As said above, each dataset admittedly focuses on one network class yet nonetheless covers a variety of network sizes and densities (Fig. 3), thus helping the task of discriminating graphs within a dataset, as is commonplace in the literature. To encompass several orders of magnitude of basic network quantities (i.e., nodes and edges), we first produce a grid of (n, m) pairs where the number of nodes n takes 15 values on a logarithmic scale from 10 to $10^{4.5}$ and the number of edges m takes 14 values on a logarithmic scale from 10 to $10^{4.25}$, while imposing that network density $\frac{2m}{n(n-1)}$ remains above 0.001 (density belongs by construction to $[0, 1]$ and equals 1 only in the case of a complete graph). This results in a grid made of 83 (n, m) pairs. Complementarily to densities, we note that we could discuss the grid in terms of ranges of average degrees $\frac{m}{n}$—whereby the corresponding contours in log-log space are still straight lines, of slope $\frac{1}{2}$, along which several datasets such as REDDIT-BINARY or DD are remarkably concentrated. Admittedly, for very large graphs in excess of a million nodes, the chosen lower bound of a density of 0.001 translates already

Table 2. Classification F1-scores over most used datasets.

	(n, m)	h^{deg}	GraphMAE	PGCL
PROTEINS	73.0 ± 4.3	74.9 ± 3.5	75.30 ± 0.39	75.7 ± 0.2
IMDB-BINARY	70.3 ± 1.7	73.1 ± 3.1	75.52 ± 0.66	-
MUTAG	87.8 ± 5.8	85.6 ± 5.9	88.19 ± 1.26	91.1 ± 1.2
NCI1	62.7 ± 2.8	68.2 ± 1.7	80.40 ± 0.30	78.8 ± 0.8
IMDB-MULTI	47.9 ± 2.9	50.9 ± 2.2	51.63 ± 0.52	-
COLLAB	68.4 ± 1.7	80.8 ± 1.0	80.32 ± 0.46	76.0 ± 0.3
DD	75.5 ± 3.4	75.2 ± 3.1	-	-
REDDIT-BINARY	83.3 ± 2.3	82.7 ± 2.4	88.01 ± 0.19	91.5 ± 0.7

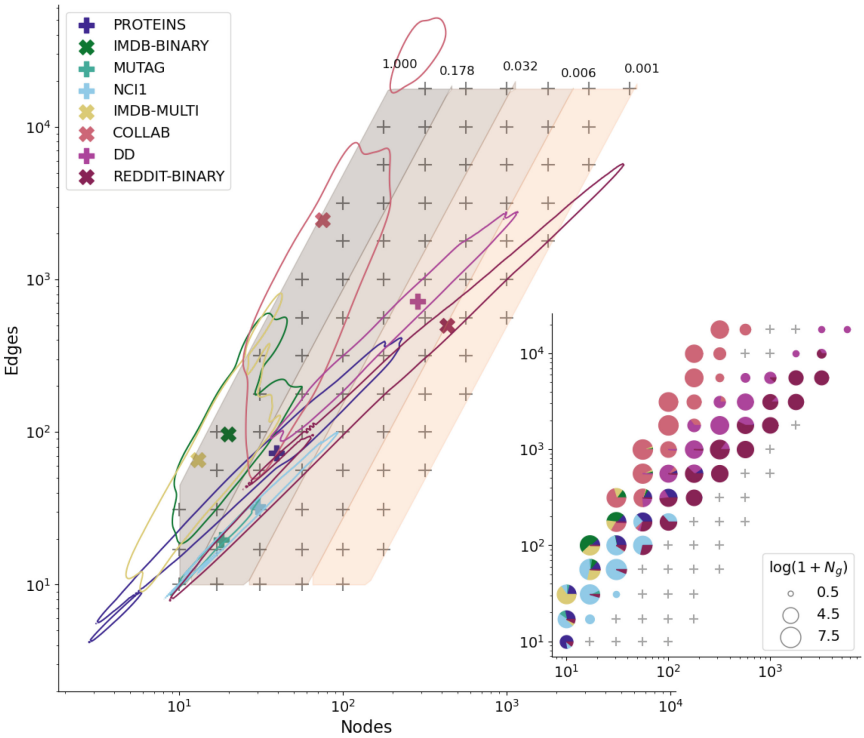


Fig. 3. Graph space in terms of number of nodes n and of edges m, in logarithmic scale. Straight lines represent isolines of graph densities (solid-colored areas delimit regions of similar density).Thin crosses '+' denote (n, m)-value pairs used to build the artificial benchmark grid.The position of popular empirical datasets in this space is figured by their barycenters $(\langle n \rangle, \langle m \rangle)$ and their extension as a kernel density estimate-based contour plot gathering 95% of the graphs, the shape of barycenter markers denotes their empirical context: '+' for natural science and '**x**' for social science. The inset plot further shows pie charts featuring the distributions of the most popular datasets in each of the synthetic grid region, their size is proportional to the logarithm of the number of graphs in the closest vicinity of each region.

into an average degree of at least 500; a value that remains credible for various social networks but that lies nonetheless at the boundary of what is realistic. A more comprehensive grid should endeavor at exploring regions of lower densities, yet it is left for further research as we limit ourselves here to the regions that correspond most to those covered by the standard empirical datasets of the GNN/GRL literature.

To further embrace several of the canonical network generation principles proposed or discovered in the literature, we leverage a generic method based on *generators* [14]. This approach allows the construction of random graphs of given size (n, m) through the iterative addition of m edges, starting with an empty network of n nodes and following a generalized preferential attachment principle configured by a generator w. The generator is a mathematical function w that assigns a non-negative weight $w(i, j)$ to all pairs of nodes (i, j). At each step, an edge is added whose extremities i and j are chosen with a probability proportional to $w(i, j)$. Generators may take into account any property of a node pair (i, j), including degrees k_i and k_j, topological distance $d(i, j)$ (the number of hops from i to j), or plain labels i or j. In the present paper, we limit ourselves to four canonical generators that are found to appear recurrently in many families of graphs stemming from a wide range of semantically different generative processes [5,14,15]: uniform link creation (Erdős-Rényi, denoted `ER`), simple preferential attachment based on degree (popularized in [2] and denoted `PA`), proximity-based attachment (denoted `D` for 'distance'), and node class-based (inspired by [15] and denoted `ID`, for it is simply based on node ID numbers from which node classes may emerge). In generator terms, this translates straightforwardly into $w_{\texttt{ER}}(i, j) = 1$, $w_{\texttt{PA}}(i, j) = k_i$, $w_{\texttt{D}} = d(i, j)$ and $w_{\texttt{ID}}(i, j) = i$. Again, a much richer portfolio of network generation principles could easily be used and tested to appraise the discriminative power of GRLs in the context of more sophisticated yet standard generative models such as stochastic block models (SBMs) [1,6], exponential random graph models (ERGMs) [19], or varieties thereof with controllable planted structure [4,13]. Our work is nevertheless meant as a proof-of-concept aimed at uncovering issues and limitations with canonical white box generative principles that remain relevant and palatable for a wide range of network sizes, including small ones; we leave to further research the task of appraising extensively the effects of choosing such or such family of generating principles in such or such regions.

For each grid point i.e., pair (n, m), we then produce a set of 64 artificial graphs for each variety of canonical generation principles i.e., for each of the four network families `ER`, `PA`, `D` and `ID`. This process yields 256 networks per grid point and, hence, the generation of 21 248 synthetic graphs with sizes varying from $(10, 10)$ to $(5\,623, 17\,782)$. As opposed to individual datasets, this approach allows to generate a set of networks covering, in a regular manner, a much thicker portion of the space of graphs, as shown on Fig. 3.

3.3 Evaluation Protocol

While the generator-based approach is able to provide any number of instances of same-sized graphs stemming from the same generating family, it concurrently serves to precisely design the downstream graph classification task, where labels are associated by construction with each graph family based on their underlying generative process. In practice, training and classification is undertaken on each point of the grid, providing 83 different classification tasks for each graph size point (n, m). This means that discrimination between the four labels occurs within a fixed pair (n, m), instead of across a more or less spread range of such values. Similarly to the most common framework adopted in the literature[*inter alia*[7,10,23,30]], the performances of different methods is measured by computing the classification F1-score of the output of a SVC classifier fed with the vector representations, or embeddings, of the graphs, averaged over 10 random folds.

As a proof of concept, we chose to evaluate the benchmark of synthetic networks using two self-supervised GRL methods: the generative method GraphMAE [7] and the contrastive method PGCL [10]. We decided to focus on self-supervised methods as we believe that such models could suit a large number of practitioners to use off-the-shelf methods on a wide array of applications without relying on extensive labeling procedures. Moreover these models appear as strong representative of the two major paradigms in GRL [11,12,24,26], namely predictive and contrastive, respectively with GraphMAE and PGCL. Plus, these specific models were reportedly among the models achieving state-of-the-art performances by the time of the our experiments. Yet, it is important to note that the parameters of the models are not tuned for the task but chosen a priori. For GraphMAE, parameters correspond to the released configuration for the `IMDB-BINARY` dataset. For PGCL, parameters are set as much as possible to the "default" values based on the published article and code.

In order to assess the difficulty of the tasks and to contextualize the performances of the tested GNNs, a straightforward and highly interpretable graph representation baseline is used. This elementary, almost trivial, baseline represents graphs by their (truncated) degree distribution. In practice, the vector used for downstream classification is the graph degree histogram. Its size is set to 64 to ease comparisons and keep a low-dimensional vector, knowing that the support of degree histograms may theoretically span $[0, n]$. As a result, the last coefficient of the vector is set to the count of nodes of degree $k \geq 64$.

4 Results

Classification F1-scores obtained with each of the three GRL methods (GraphMAE, PGCL, and "degrees") are presented in Fig. 4. Strikingly, the three figures display comparable profiles. Classification accuracies are very low for small graphs, which is not unexpected: indeed, in this region of the network space, the generative process likely considers too few edges added in order to synthesize structurally different graphs. The generative process could also explain, at

least partly, why very sparse and, dually, very dense regions of the space exhibit lower performances. On the contrary, all of the tested models reach excellent and often perfect scores on large graphs with densities approaching 1% and an average degree roughly in the order of magnitude of 10^1.

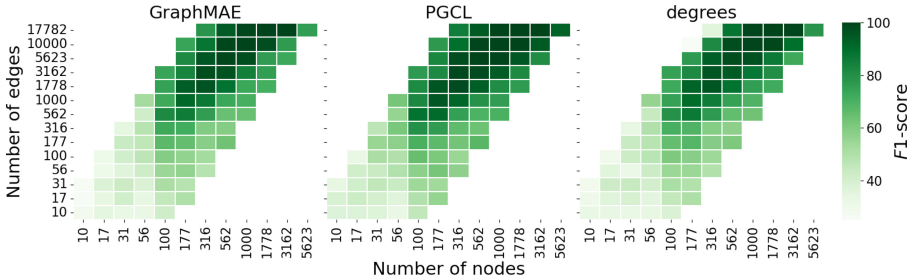


Fig. 4. F1-score of classification tasks over the synthetic graph grid (x-axis: number of nodes, y-axis: number of edges).

The surprisingly good performance of degree histograms —a rather trivial, yet rarely benchmarked, graph representation approach— led us to contrast the deep learning methods with respect to this baseline in Fig. 5. This comparison first highlights the better performances of the tested version of PGCL over GraphMAE on this benchmark. More importantly, it appears that both deep learning models strongly outperform the degree-based baseline on large dense graphs only, where the cut-off of the degree histogram is also likely to induce the most information loss. Notwithstanding, GraphMAE is still beaten by the simple degree baseline both for small dense graphs and sparse large graphs. On the other hand, PGCL displays almost uniformly better results than the degree histogram, although the gap is not often very sizable and exhibits a remarkable variance across the network size landscape.

Link with real-world datasets. The appraisal, through the lens of the TAB, of relatively state-of-the-art GRL methods that rely on two complementary types of approaches within the GNN landscape, enables us to suggest that the added value of a heavy computational pipeline is likely limited, perhaps even suboptimal, for a large portion of the empirical graph space, precisely where many datasets used in the literature lie (this is further visible by comparing Figs. 3 and 5). On the whole, we argue that the TAB may both reveal which regions of the network size space such or such model is most efficient for —and, correspondingly, where it is least likely to be useful— and should help avoiding the possibly peculiarities and local biases induced by choosing a set of mainstream benchmarks. In any extent, this type of thick artificial benchmark should certainly be included in further works aimed at establishing the state-of-the-art performance of a GRL model, along with empirical datasets.

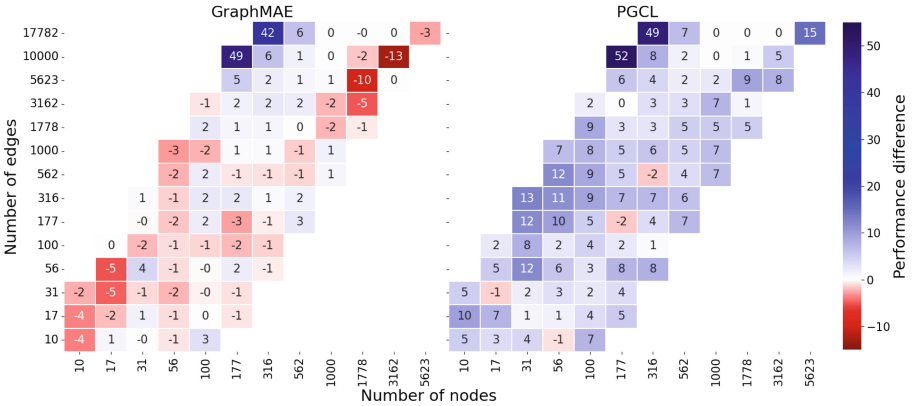


Fig. 5. Difference between the F1-score obtained with the deep learning methods (*left:* GraphMAE, *right:* PGCL) and the degree-based baseline. Red cells (negative values) indicate that the degree histogram outperforms the deep learning model, blue cells (positive values) indicate the contrary.

5 Concluding Remarks

This work presents several important contributions to the GRL field, particularly in its evaluation processes. We demonstrate that state-of-the-art models, while powerful, may not perform optimally across all types of graphs, indicating that their generalization capabilities can be task-specific. To address this, we propose a systematic and easily extendable synthetic benchmark framework that spans a wide range of graph properties, offering a more comprehensive alternative to the limited empirical benchmarks that dominate current evaluations. This framework allows us to expose regions of the graph space where naive, simple methods—such as using degree distributions—can outperform more computationally intensive GRL models. This observation suggests that the added complexity of deep learning methods does not uniformly lead to improved performance, but rather offers value only in specific regions of the graph space. Our core proposal highlights the inadequacy of existing benchmarks for evaluating GRL methods, even in basic classification tasks, and underscores the need for broad, tunable benchmarks that can more effectively assess model performance across diverse graph properties. Our approach has nonetheless several limitations that we leave for further research, mainly related to the synthetic network generation method, whose scalability and thoroughness could be improved (dealing with larger graphs and more diverse network generation families) and the graph representation learning per se, whose space of hyperparameters and tasks (beyond classification) could be explored more thoroughly, and whereby transfer learning abilities could also ultimately be leveraged (i.e., by exploring the impact of using TAB networks to train GRL methods). By appraising state-of-the-art GRL approaches against simpler, white-box methods, we provide clarity on where GRL truly adds value. As the GRL field continues to expand rapidly,

we advocate for the adoption of synthetic benchmarking frameworks that can capture a more diverse and representative segment of the network space, ultimately leading to more robust and generalizable GRL models.

Ackowledgments. This work has been partially supported by the "Socsemics" ERC Consolidator grant (grant agreement No. 772743) and the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 945304—Cofund AI4theSciences hosted by PSL University. We thank the anonymous reviewers for fruitful feedback and clarification suggestions.

References

1. Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P.: Mixed membership stochastic blockmodels. J. Mach. Learn. Res. **9**, 1981–2014 (2008)
2. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**, 509–512 (1999)
3. Borgwardt, K.M., Ong, C.S., Schönauer, S., Vishwanathan, S., Smola, A.J., Kriegel, H.P.: Protein function prediction via graph kernels. Bioinformatics **21**(suppl_1), i47–i56 (2005)
4. Cazabet, R., Citraro, S., Rossetti, G.: Structify-net: random graph generation with controlled size and customized structure. Peer Commun. J. **3** (2023)
5. da Fontoura Costa, L., Rodrigues, F.A., Travieso, G., Boas, P.R.V.: Characterization of complex networks: a survey of measurements. Adv. Phys. **56**(1), 167–242 (2007)
6. Holland, P.W., Laskey, K.B., Leinhardt, S.: Stochastic blockmodels: first steps. Social Netw. **5**, 109–137 (1983)
7. Hou, Z., et al.: Graphmae: self-supervised masked graph autoencoders. In: 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 594–604 (2022)
8. Keramatfar, A., Rafiee, M., Amirkhani, H.: Graph neural networks: a bibliometrics overview. Mach. Learn. Appl. **10**, 100,401 (2022)
9. Li, P., et al.: Pairwise half-graph discrimination: a simple graph-level self-supervised strategy for pre-training graph neural networks. arXiv:2110.13567 (2021)
10. Lin, S., et al.: Prototypical graph contrastive learning. IEEE Trans. Neural Netw. Learn. Syst. (2022)
11. Liu, X., et al.: Self-supervised learning: generative or contrastive. IEEE Trans. Knowl. Data Eng. **35**(1), 857–876 (2021)
12. Liu, Y.: Graph self-supervised learning: a survey. IEEE Trans. Knowl. Data Eng. **35**(6), 5879–5900 (2022)
13. Mangold, L., Roth, C.: Generative models for two-ground-truth partitions in networks. Phys. Rev. E **108**(5), 054,308 (2023)
14. Menezes, T., Roth, C.: Symbolic regression of generative network models. Sci. Rep. **4**(1), 6284 (2014)
15. Menezes, T., Roth, C.: Automatic discovery of families of network generative processes. In: Ghanbarnejad, F., Saha Roy, R., Karimi, F., Delvenne, J.-C., Mitra, B. (eds.) DOOCN 2017. SPC, pp. 83–111. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-14683-2_4

16. Milo, R., et al.: Superfamilies of evolved and designed networks. Science **303**(5663), 1538–1542 (2004)
17. Morris, C., Kriege, N.M., Bause, F., Kersting, K., Mutzel, P., Neumann, M.: Tudataset: a collection of benchmark datasets for learning with graphs. In: ICML 2020 Workshop GRL+ 2020 (2020)
18. Onnela, J.P., et al.: Taxonomies of networks from community structure. Phys. Rev. E **86**, 036,104 (2012)
19. Robins, G., Pattison, P., Kalish, Y., Lusher, D.: An introduction to exponential random graph (p*) models for social networks. Social Netw. **29**(2), 173–191 (2007)
20. U.S. Department of Health and Human Services: AIDS antiviral screen data - NCI DTP data - NCI Wiki. https://wiki.nci.nih.gov/display/NCIDTPdata/AIDS+Antiviral+Screen+Data
21. Wei, S., Zhao, Y.: Graph learning: a comprehensive survey and future directions. arXiv preprint arXiv:2212.08966 (2022)
22. Wen, Q., Ouyang, Z., Zhang, C., Qian, Y., Ye, Y., Zhang, C.: Graph contrastive learning with cross-view reconstruction. In: NeurIPS 2022 Workshop: New Frontiers in Graph Learning (2022)
23. Wu, J., Chen, X., Shi, B., Li, S., Xu, K.: Sega: structural entropy guided anchor view for graph contrastive learning. In: International Conference on Machine Learning, pp. 37,293–37,312. PMLR (2023)
24. Wu, L., Lin, H., Tan, C., Gao, Z., Li, S.Z.: Self-supervised learning on graphs: contrastive, generative, or predictive. IEEE Trans. Knowl. Data Eng. **35**(4), 4216–4235 (2023)
25. Wu, Z.: Moleculenet: a benchmark for molecular machine learning. Chem. Sci. **9**(2), 513–530 (2018)
26. Xie, Y., Xu, Z., Zhang, J., Wang, Z., Ji, S.: Self-supervised learning of graph neural networks: a unified review. IEEE Trans. Pattern Anal. Mach. Intell. **45**(2), 2412–2429 (2022)
27. Yanardag, P., Vishwanathan, S.V.N.: Deep graph kernels. In: 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2015)
28. Yang, L., Hong, S.: Omni-granular ego-semantic propagation for self-supervised graph representation learning. In: International Conference on Machine Learning, pp. 25,022–25,037. PMLR (2022)
29. Yaveroglu, O.N., et al.: Revealing the hidden language of complex networks. Scientific Reports **4**(4547) (2014)
30. You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., Shen, Y.: Graph contrastive learning with augmentations. Adv. Neural. Inf. Process. Syst. **33**, 5812–5823 (2020)

Dissecting Embedding Method: Learning Higher-Order Structures from Data

Liubov Tupikina[1,2,3](✉) and Hritika Kathuria[2]

[1] Nokia Bell labs, France, Paris, France
liubov.tupikina@cri-paris.org
[2] LPI, University Paris Descrartes, Learning Planet Institute, INSERM, Paris, France
[3] MIPT, Moscow Institute of Physics and Technology, Moscow, Russia

1 Introduction and Problem Statement

Active area of research in AI is the theory of manifold learning and finding lower-dimensional manifold representation on how we can learn geometry from data for providing better quality curated datasets [1]. There are however various issues with these methods related to finding low-dimensional representation of the data, the so-called curse of dimensionality [2].Geometric deep learning methods for data learning often include set of assumptions on the geometry of the feature space. Some of these assumptions include pre-selected metrics on the feature space, usage of the underlying graph structure, which encodes the data points proximity. However, the later assumption of using a graph as the underlying discrete structure, encodes only the binary pairwise relations between data points, restricting ourselves from capturing more complex higher-order relationships, which are often often present in various systems [1,5]. These assumptions together with data being discrete and finite can cause some generalisations, which are likely to create wrong interpretations of the data and models outputs. Hence overall this can cause wrong outputs of the embedding models themselves, while these models being quite and trained on large corpora of data, such as BERT, Yi and other similar models.

The objective of our research is twofold, first, it is to develop the alternative framework to characterize the embedding methods dissecting their possible inconsistencies using combinatorial approach of higher-order structures which encode the embedded data. Second objective is to explore the assumption of the underlying structure of embeddings to be graphs, substituting it with the hypergraph and using the hypergraph theory to analyze this structure. We also demonstrate the embedding characterization on the usecase of the arXiv data.

2 Data

The arXiv dataset encompasses scientific articles published online between 1992 and 2018, which span 175 scientific fields [3]. Each article has the ground-truth information (metadata) stored as a category or a tag, e.g. *cond-mat.dis-nn* and others.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 369–373, 2025.
https://doi.org/10.1007/978-3-031-82439-5_30

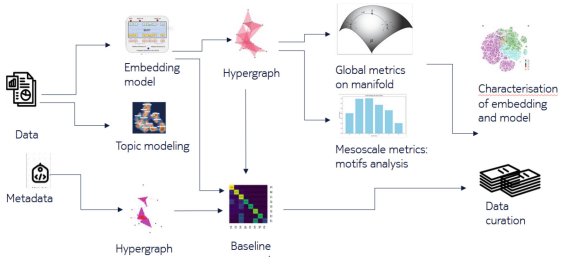


Fig. 1. Scheme of the methodological framework.

3 Methodological Framework

Our methodological framework consists of the following steps, Fig. 1. **First**, we apply the embedding model to the datapoints D_i with the metadata m_i to get the coordinates of the embedded datapoints. **Second**, we cluster the datapoints D_i in the embedding using some clustering method (e.g. k-means clustering method) and apply ML metrics (e.g. confusion matrix, accuracy scores) to deduce scores of the embedding model, which produced the vector representation of datapoints D_i using the metadata labels of clusters m_i.**Third,** we test the discripancies in the resulted confusion matrix by constructing the *neighborhood hypergraph* H from datapoints $\{D_i\}_{i=1}^n$ and calculate hypergraph motifs in this hypergraph H, corresponding to the embedding. Then we analyze the hypergraphs constructed from the metadata m_i and sub-hypergraphs, which contain the misclassified tags based on the metrics from the second step of the method. Motifs in hypergraphs are subhypergraphs structures. By studying them one can dissect the most popular of the higher-order intricate properties of data structures [7].

Neighborhood Hypergraph Construction Method. The *neighborhood hypergraph* H_n is constructed from data points $\{D_i\}_{i=1}^n$ based on the idea of estimating the relations between the datapoints. If datapoints $(D_{i_1}, ..., D_{i_k})$ are close to each other, they form a hyperedge of a hypergraph in H_n of k-arity. Depending on the nature of data there are different ways to characterize closseness of datapoints, which can be based on their geometrical or semantical proximity. Geometrical proximity of points $\{D_i\}$ is estimated using typical $l2$ metrics or cosine similarity applied to coordinates of each processed datapoint D_i. Semantical proximity of points is encoded by labelled similarity of datapoints, for instance, in case of arXiv data it means that if paper $\{D_1\}$ has tags a, b, c, paper D_2 has tags a, b and paper D_3 has tags a, d, then these papers are encoded as nodes of the hypergraph forming the hyperedge with the field a. One can also construct other encoding of the higher-order structure from data using *metadata* $\{m_i\}_{i=1}^m$ information H_m which encodes relations between tags themselves and can characterize the fields closseness instead of using the standard binary graph representation of data as co-tags networks [3]: when tags a, b, c are nodes of hypergraph, these nodes share the hyperedge, iff there is at least one paper

with all these tags, like paper D_1 in this case. Our intuition behind using the *neighborhood hypergraph* representation of data is based on the observation that in many textual data similarities between documents are not just based on one measure of closeness but rather on high-order relations like in words: word *France* is close to *Italy* and *Slovenia* in different way than *France* to the word *Paris*.

The Embedding Models. In this short paper we choose BERT as **the embedding model**, general autoencoder architecture, which provides the embedding of the textual data [2]. Our framework can be applied to other embedding models.

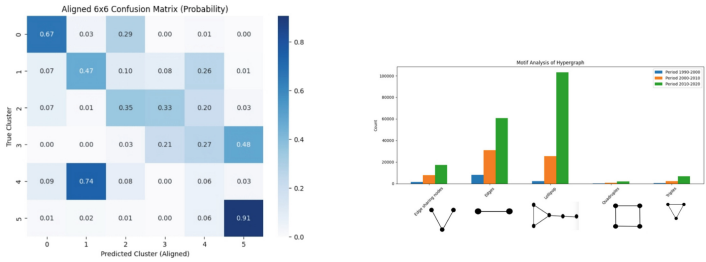


Fig. 2. (Left) Confusion matrix is calculated for the clustered datapoints from BERT embedding method to compare them to the clusters from ground truth labels (metadata).We consider subsample of data from (2000–2011). (Right) Characterisation of persistent hypergraph motifs in the hypergraphs constructed from abstracts datapoints D_i for each decade.

Analytical Background of the Method and General Relation to the Manifold Learning. Our general motivation comes from the manifold regularisation method, which allows one to learn the properties of data limiting the space of solutions to the manifold. The central idea of the manifold learning is to form a neighborhood graph G_n [1].We aim further to extend the idea of considering the neighborhood hypergraph H_n instead of the neighborhood graph as the underlying learned structure from data. Usually we assume that the feature space of data has the standard $l2$ metric, which induces the Hilbert space. The neighborhood graphs (hypergraphs) are constructed such that one chooses k nearest points as the neighborhood or points at the maximum distance r for the neighborhood.Manifold regularization further simplifies assumptions using a binarized metric or heat kernel. The problem of the manifold regularisation with the hypergraph H_n then can be further formalized as:

$$f^* = \min_{f \in h_k} (L(f)) + \lambda(f^T L(H_n) f),$$

where $L(H_n)$ is the generalized hypergraph Laplacian. $(f, L(H_n)f) = f^T L(H_n) f$ is a scalar product, λ is the regularization parameter [1], f^T is the transpose of a

vector, where $\min_{f \in h_k}$ is minimum found from the set of possible hypothesis e.g. hypothesis on spaces search. Finding the underlying hypergraph can then extend the general method for manifold learning from data, yet it has some drawbacks, such as increasing complexity of the computations.

4 Results

We cluster datapoints D_i with coordinates estimated by BERT method. Then we estimate the confusion matrix, Fig. 2 **(left)**, to characterize the potential discrepancy in accuracy of the embedding method, verifying the clustered datapoints according to the labels of metadata m_i. Most of the misclassified datapoints belong to the cluster with *condmat* metadata field label, which is also known to be interdisciplinary field [3]. Majority of papers in *condmat* typically share common tags with other fields, which correspond to its propensity to be involved in various hyperedges of the *neighrborhood hypergraph* reconstructed from the data. For exploring this aspect we analyze the hypergraph, which encodes the proximity of abstracts (our datapoints D_i). For this we study the higher-order motifs in the whole hypergraph and its subhypergraphs, Fig. 2 **(right)**. We studied both, hypergraph H_n constructed from all datapoints D_i, as well as subhypergraphs, which consists of only nodes with *condmat* as their tags. In those subhypergraphs we found higher-order motifs, such as *triples, lollipops* in both structures, and we were able quantitatively characterize them. Moreover, we see the persistent growth in number of such higher-order motifs from the first decade of arXiv data to the present one, and specifically lollipop motifs, which are the most persistent patterns in these hypergraphs.

5 Conclusions

In our work we develop the framework to dissect the geometrical properties of embeddings based on the higher-order structures. This approach can be used in addition to traditional combinatorics and information theory based approaches to characterize embeddings [4]. In particular, in our approach we characterize possible inconsistencies in embedded data estimating the prevailing motifs in subhypergraphs, associated with the misclassified data. We plan to incorporate hypergraph embedding methods [6] further to investigate their applicability to data curation for LLMs training, as well as to dimensionality reduction methods. Currently we studied only BERT embedding model, yet our methodology has the potential to be further extended to differentiate between distinct embeddings, when we construct corresponding hypergraphs to each separate embedding method and then compare embeddings using the hypergraphs theory [7].

References

1. Melas-Kyriazi, L.: Mathematical foundations of manifold learning (2020)
2. de Dampierre, C., Mogoutov, A., Baumard, N.: Towards transparency: exploring LLM trainings datasets through visual topic modeling and semantic frame. arXiv (2024)
3. Singh, C., Tupikina, L., Starnini, M., Santolini, M.: Charting mobility patterns in the scientific knowledge landscape. EPJ Data Sci. **13**, 12 (2024)
4. Grootendorst, M.: BERTopic: neural topic modeling with a class-based TF-IDF procedure. arXiv preprint arXiv:2203.05794
5. Battiston, F., et al.: Networks beyond pairwise interactions: structure and dynamics. Phys. Rep. **874** (2020)
6. Shi, F., Evans, J.: Surprising combinations of research contents and contexts are related to impact and emerge with scientific outsiders from distant disciplines. Nat. Commun. **14**, 1641 (2023)
7. Zapata-Carratala, C., et al.: Hypermatrix algebra and irreducible arity in higher-order systems. Adv. Compl. Syst. (2023)

Local Intrinsic Dimensionality for Dynamic Graph Embeddings

Dušica Knežević, Miloš Savić(✉), and Miloš Radovanović

Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 3, 21000 Novi Sad, Serbia
{lucy,svc,radacha}@dmi.uns.ac.rs

Abstract. The notion of local intrinsic dimensionality (LID) has important theoretical implications and practical applications in the fields of data mining and machine learning. Recent research efforts indicate that LID measures defined for graphs can improve graph representational learning methods based on random walks. In this paper, we discuss how NC-LID, a LID measure designed for static graphs, can be adapted for dynamic networks. Focusing on dynnode2vec as the most representative dynamic graph embedding method based on random walks, we examine correlations between NC-LID and the intrinsic quality of 10 real-world dynamic network embeddings. The obtained results show that NC-LID can be used as a good indicator of nodes whose embedding vectors do not tend to preserve temporal graph structure well. Thus, our empirical findings constitute the first step towards LID-aware dynamic graph embedding methods.

Keywords: dynamic graphs · graph embeddings · local intrinsic dimensionality

1 Introduction

Many real-world complex systems studied in diverse research and engineering disciplines can be modeled by graphs or networks showing relations or interactions among their constituent units [11]. Such complex systems are typically dynamic, meaning that the structure of underlying networks evolves in time by adding and deleting both nodes and edges [3]. Networks representing complex dynamical systems are also called *dynamic* or *temporal networks*.

Graph-structured data describing complex dynamical systems can be used to construct various types of useful predictive models relying on machine learning (ML) algorithms and methods [1,5]. Typical examples include models for node classification, node clustering, link prediction, node/link attribute prediction, anomaly detection, diffusion prediction, etc. There are three different approaches to build such predictive models: (1) by graph-native ML techniques (e.g., iterative classification, community detection), (2) by applying ML algorithms designed for tabular data either to hand-crafted features or features learned by graph embedding algorithms, and (3) by graph neural networks [14].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 374–385, 2025.
https://doi.org/10.1007/978-3-031-82439-5_31

The focus of this paper is on dynamic graph embedding algorithms. In contrast to graph neural networks showing superior performance when building task-specific predictive models for feature-rich attributed graphs, graph embedding algorithms provide a task-agnostic approach for learning general-purpose graph representations of feature-sparse and non-attributed networks. The most numerous, scalable and robust general-purpose dynamic graph embedding methods are ones based on random walks [1,5]. The main idea of these methods is to sample a certain number of random walks for each node that are then treated as sentences over node identifiers. In this way, the problem of generating graph embeddings is reduced to the problem of generating text embeddings.

In our previous works [6,12,13], it has been shown that static graph embedding methods based on random walks can be improved by taking into account local intrinsic dimensionality (LID) [4] aspects of nodes. In [12] we have proposed a principal approach for formulating LID-related measures for nodes in static graphs and defined one concrete LID measure NC-LID that is based on natural communities [7] as intrinsic node localities. Then, we have designed and evaluated two LID-aware node2vec [2] extensions based on the NC-LID measure showing that they improve intrinsic embedding quality [13], as well as the performance of concrete machine learning applications [6].

In this work, we discuss how NC-LID can be adapted for dynamic networks (Sect. 2). As the main contribution, we propose a methodology to evaluate correlations between NC-LID and the intrinsic quality of dynamic graph embeddings (Sect. 3). The application of the proposed methodology to one concrete dynamic graph embedding method (dynnode2vec [8]) and various real-world dynamic networks reveal that NC-LID is able to point to nodes with poorly constructed embedding vectors that do not preserve local connectedness of the corresponding nodes (Sect. 4). Therefore, just as for static graph embedding methods, these results indicate that appropriate LID-aware extensions, outlined in the last section of the paper (Sect. 5), have the potential to improve dynamic graph embedding methods.

2 Local Intrinsic Dimensionality Measure for Dynamic Networks

The intrinsic dimension (ID) d' of a set of data points D in a d-dimensional Euclidean space can be intuitively understood as the minimal number of features required to represent D in a lower-dimensional space ($d' < d$) without significant information loss, or as the dimension of the surface achieving the best fit to D. However, ID may vary across a dataset leading to the notion of local intrinsic dimensionality (LID), with the main idea to asses ID of local data spaces surrounding individual data points. In his seminal paper, Houle [4] used the ball-expanding model to formally define LID w.r.t. the distribution of distances to a reference data point p. Intuitively, LID expresses the degree of difficulty to separate nearest neighbors of p from the rest of the dataset, thus

reflecting the degree of indiscriminability of the underlying distance function in the vicinity of p.

The works by Savić et al. [12,13] were the first attempts to define LID measures for nodes in a graph. Building on Houle's LID foundations for tabular datasets, Savić et al. proposed an approach for formulating LID-related measures for graphs following the idea of substituting a ball around a data point with a subgraph around a node in order to be able to estimate the discriminatory power of a graph-based distance of interest. A measure called NC-LID was introduced for the shortest-path distance as the underlying distance function and natural or local communities [7] of nodes as their intrinsic localities.

A community in a graph is a highly cohesive and dense subgraph sparsely connected to the rest of the graph [11]. The natural community of node n is a community recovered from n as the seed node. NC-LID relies on natural communities identified by the fitness-based algorithm proposed by Lancichinetti et al. [7]. Let S be the natural community of n in an unweighted graph $G = (V, E)$, and let k denote the largest shortest-path distance from n to any node in S. The NC-LID of n is then formally defined as:

$$\text{NC-LID}(n) = -\ln\left(\frac{|S|}{D(n,k)}\right),$$

where $|S|$ is the number of nodes in S and $D(n, k)$ is the number of nodes whose shortest-path distance from n is smaller than or equal to k [12,13]. The lowest possible value of NC-LID is equal to 0. In such cases, the shortest-path distance can perfectly separate S from the rest of the nodes in G, i.e., S encompasses all nodes that are maximally k hops away from n. Higher values of NC-LID(n) imply that it is harder to distinguish S from the rest of G relying on the shortest-path distance. This means that with higher NC-LID natural communities take more complex, irregular shapes. It is easy to see that NC-LID values of nodes in weighted graphs can be obtained by (1) using an appropriate distance measure instead of the shortest-path distance (e.g., Dijsktra's distance), and (2) recovering natural communities relying on the fitness function that uses node strength instead of node degree.

The NC-LID measure defined for static graphs can be adapted for dynamic graphs in two principal ways depending on the chosen temporal graph representation model. In discrete-time representation models, the evolution of a dynamic graph is described by a sequence of static graphs that are called snapshots. Thus, for a node we have a sequence of NC-LID values considering those snapshots in which the node appears. Since dynamic graph algorithms designed for discrete-time representations incrementally operate on static graphs, the original NC-LID definition for static graphs can be used without any modifications when designing their LID-aware counterparts.

The evolution of a graph in continuous-time representation models is given by a sequence of events that correspond to interactions between nodes. Events are usually represented by triplets in the form (x, y, t), where x and y are two nodes involved in the interaction that occurs at time t. Additionally, the duration of

interactions can be specified if interactions are not instantaneous.[1] Algorithms designed for continuous-time representation models usually operate on a single graph that aggregates all the events. Edges in that aggregated graph are time-stamped, i.e., each edge is attributed with time points or time intervals at which the corresponding interaction occurs. In such setting, node distances can be defined considering temporal or time-varying paths [10]. More specifically, for two events (x, y, t_1) and (y, z, t_2) there is a temporal path $x \rightarrow y \rightarrow z$ if $t_2 > t_1$. The distance between two nodes i and j at time t can be defined as the shortest time it takes to reach j from i at time t along temporal paths. The temporal distance between i and j can then be obtained by averaging time-dependent distances over the entire period (from the first to the last event). Consequently, the NC-LID of a node can be obtained by the static graph definition in which the shortest-path distance is replaced by the temporal distance and without any modification regarding natural community detection.

3 Evaluation Methodology

The aim of this work is to evaluate the ability of the NC-LID measure to improve dynamic graph embedding algorithms based on random walks. We consider discrete-time representations of dynamic graphs, since most of the existing dynamic graph embedding methods are based on this approach [1].

To obtain embeddings of real-world dynamic graphs we use dynnode2vec [8]. However, the methodology described in this section can be applied to other dynamic graph embedding methods as well. Dynnode2vec is an adaptation for dynamic graphs of the most representative random-walk based static graph embedding method – node2vec [2]. As the input dynnode2vec accepts an evolutionary sequence of graph snapshots. Similarly as node2vec, embedding construction is based on biased random walks, whose sampling is controlled by four parameters: the number of random walks sampled from each node, the length of random walks, the return parameter p controlling the probability of going back to the previous node in the walk, and the in-out parameter q controlling the probability of exiting the immediate neighborhood of the previous node. Using the first snapshot in the input sequence, dynnode2vec samples random walks to construct an initial skip-gram model, performing both steps in the same way as node2vec. The initial model is then incrementally updated with each subsequent snapshot by random walks sampled from the so-called evolutionary nodes, i.e., nodes whose neighborhoods have changed in two consecutive snapshots. Node embedding vectors are constructed from the skip-gram model after each model update, which means that dynnode2vec provides embeddings for each individual snapshot in the evolutionary sequence.

The main goal of general-purpose graph embedding methods is to preserve structural properties of networks in embedded spaces. Thus, well performing graph embedding methods need to result with embeddings from which original

[1] It should be emphasized that in the most of publicly available datasets the duration of interactions is not provided.

graphs can be reconstructed [13]. The graph reconstruction process is performed by computing the Euclidean distance between embedding vectors for each pair of nodes and then connecting the $|E|$ closest node pairs, where $|E|$ is the number of links in the original graph. Let n denote an arbitrary node and let C be the number of correctly reconstructed links incident to n. Then, the following embedding evaluation metrics can be computed:

- Precision – C divided by the number of links n has in the reconstructed graph.
- Recall – C divided by the number of links n has in the original graph.
- F_1 score – the harmonic mean of precision and recall.

Higher values of precision, recall and F_1 imply a lower number of link reconstruction errors for n. Precision, recall and F_1 at the graph level can be obtained by micro-averaging over the nodes.

In our methodology, dynnode2vec is tuned by finding values of its hyper-parameters p and q that give embeddings maximizing the F_1 score in the last snapshot for five different graph embedding dimensions (10, 25, 50, 100 and 200). For p and q we consider values in $\{0.25, 0.50, 1, 2, 4\}$, while the number of random walks per node and the length of each random walk are fixed to 10 and 32, respectively. For one configuration of hyper-parameters, the F_1 score is determined by averaging the results of 10 runs.

To examine how NC-LID is related to the quality of dynamic graph embeddings, we compute Spearman correlations between NC-LID and above defined embedding evaluation metrics over all graph snapshots. Significant negative correlations between NC-LID and embedding evaluation metrics indicate that NC-LID can point to nodes that have poorly constructed embedding vectors. Additionally, throughout the whole evolutionary sequence, nodes are assigned to two categories:

1. nodes having NC-LID above the average NC-LID in the current snapshot (denoted by H), and
2. nodes having NC-LID below the average NC-LID in the current snapshot (denoted by L).

Then,F_1 scores of those two groups of nodes are compared by the Mann-Whitney U (MWU) test [9]. MWU is a test of stochastic equality checking the null hypothesis that F_1 scores in H do not tend to be neither greater nor smaller than F_1 scores in L. Additionally, we compute two probabilities of superiority reflecting the strength of stochastic inequality:

1. PS(H) – the probability that a randomly selected node from H has a higher F_1 than a randomly selected node form L,
2. PS(L) - the probability that a randomly selected node from L has a higher F_1 than a randomly selected node from H.

Additionally, we investigate whether high NC-LID nodes tend to be the most structurally important nodes in the network. To address this question, we compute the Spreman correlation coefficient between NC-LID and widely used node centrality metrics: degree, betweenness, closeness, eigenvector and shell index centrality [11].

4 Results and Discussion

Our experimental evaluation of NC-LID is conducted on real-world dynamic networks listed in Table 1. Examined networks are typically used dynamic networks in the literature and can be retrieved from two publicly-available repositories: SNAP[2] and Network Repository.[3] Besides the total number of nodes (denoted by $|V|$) and the total number of links (denoted by $|E|$) that can be observed throughout network evolution, the number of snapshots and time resolution of each snapshot, Table 1 also shows the average activation of nodes and links, denoted by $a(V)$ and $a(E)$, respectively. Namely, the activation of a node (resp. link) is the number of snapshots in which the node (resp. link) is present or active.

Table 1. Experimental datasets.

Network	$\|V\|$	$\|E\|$	snapshots	time resolution	$a(V)$	$a(E)$
ia-hospital	75	1369	4	day	2.79	1.34
ia-contacts	113	2470	3	day	2.65	1.19
ia-enron	151	2047	11	3-months	6.57	1.74
radoslaw-email	167	4519	9	month	7.5	2.35
ia-primschool	242	9843	2	day	1.98	1.16
fb-forum	898	8212	5	month	3.03	1.28
email-eu	986	22205	17	month	12.29	2.79
college-msg	1894	14474	6	month	2.23	1.08
fb-messages	1899	16415	7	month	2.45	1.07
ia-realitycall	6763	7947	15	week	2.24	2

The average and maximal NC-LID over all snapshots for examined dynamic graphs are shown in Fig. 1, sorted from the lowest to the highest average NC-LID. It could be observed that the average NC-LID does not depend on network size and density, i.e., smaller (resp., sparser) networks may have more complex natural communities than larger (resp., denser) networks. For example, the largest average NC-LID is exibited by `fbforum`, which is 6th (out of 10 networks) by size and 4th by density.

Fig. 2 shows Sperman correlations between NC-LID and node centrality metrics. It can be seen that in 5 networks there are significant positive correlations, in 3 networks significant negative correlations, while in 1 network correlations are not strong. The strongest correlations, either positive or negative, are present for closeness centrality. This implies that, in some networks, nodes with complex

[2] https://snap.stanford.edu/data/#temporal.
[3] https://networkrepository.com/dynamic.php.

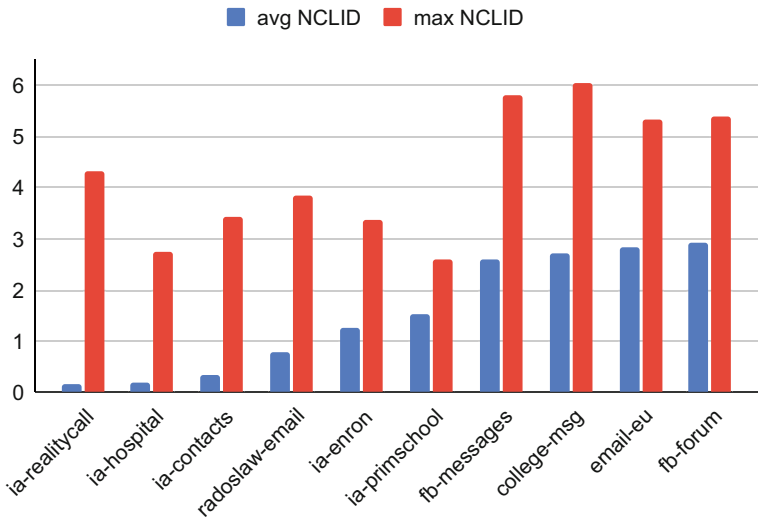


Fig. 1. Average and maximal NC-LID per network.

natural communities tend to be located at network periphery (negative correlations), while in some networks such nodes tend to occupy central positions and be the most important nodes for dynamic processes running on networks.

The results of dynnode2vec tuning are presented in Table 2. This table shows the maximal F_1 scores for dynamic networks from our experimental corpus sorted from the highest to the lowest F_1, the dimension in which the maximal F_1 score is achieved (Dim.), precision, recall and the corresponding values of p and q. It can be observed that for 9 dynamic networks (all except `ia-realitycall`) we have F_1 scores in the range $[0.56, 0.83]$. Such high F_1 scores imply that dynnode2vec embeddings preserve the structure of input dynamic networks to a great extent. Additionally, it can be seen that precision and recall tend to have close values implying that the corresponding quality attributes tend to be balanced in the best dynnode2vec embeddings. Consequently, it can be concluded that dynnode2vec achieves a very good performance at making general-purpose vector representations of nodes in dynamic networks, justifying our choice of this particular state-of-the-art dynamic graph embedding method for the purpose of NC-LID evaluation.

High NC-LID nodes have more complex natural communities than low NC-LID nodes. Thus, it is quite reasonable to expect that high NC-LID nodes will have a higher number of link reconstruction errors due to more complex natural communities. To verify this assumption, we compute Spearman correlations between NC-LID and embedding evaluation metrics (precision, recall, F_1) obtained from the best dynnode2vec embeddings. Since lower values of precision, recall and F_1 imply a higher number of incorrectly reconstructed links, counting both false positives and false negatives, significant negative correla-

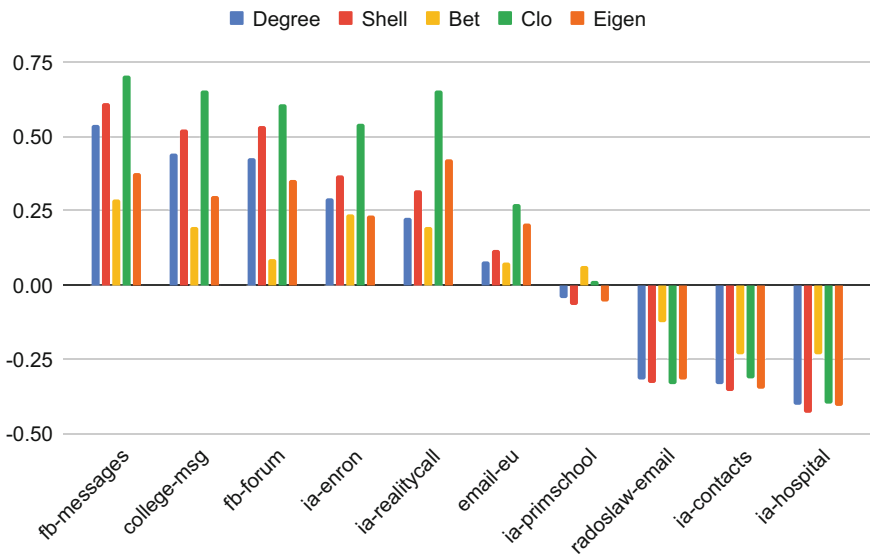


Fig. 2. Spearman correlations between NC-LID and node centrality measures.

tions between NC-LID and those embedding quality metrics would imply that NC-LID is able to point to nodes with 'weak' representations ('weak' in the sense that their embedding vectors do not preserve their connectedness in the embedding space). The values of Spearman correlations are shown in Fig. 3. It can be observed that for 8 networks there are negative Sperman correlations between NC-LID and F_1. Only for 2 networks we have positive correlations, where one of the networks is `ia-realitycall` (please recall that the dynnode2vec embedding for this particular network has extremely low precision, recall and F_1 scores, thus the correlation results obtained for this network can not be used to derive a meaningful conclusion regarding the relationship between NC-LID and embedding quality).

For the first 6 networks shown in Fig. 3 there are significant negative correlations between NC-LID and F_1 ranging from -0.19 to -0.41. For the same networks significant negative correlations between NC-LID and recall are also present (ranging from -0.19 to -0.36), and they also tend to exhibit significant negative correlations between NC-LID and precision with two exceptions (`ia-contacts` with a significant positive correlation and `radoslaw-email` without any significant correlation regarding precision). For the next two networks depicted in Fig. 3 we have significant negative correlations between NC-LID and recall. Therefore, it can be concluded that for the majority of networks in our experimental corpus there are notable negative correlations between NC-LID and at least one metric reflecting intrinsic embedding quality.

Table 2. Characteristics of the best dynnode2vec embeddings (sorted from the highest to the lowest F_1 score).

Graph	Dim.	p	q	Precision	Recall	F_1
fb-forum	200	1	1	0.8289	0.8346	0.8317
ia-primschool	10	1	4	0.8051	0.7869	0.7959
college-msg	50	1	1	0.664	0.7043	0.6835
email-eu	50	0.5	0.5	0.6325	0.7174	0.6722
ia-contacts	200	0.5	4	0.6944	0.6158	0.6527
ia-hospital	100	0.5	4	0.6476	0.6087	0.6274
radoslaw-email	25	0.5	4	0.6662	0.5925	0.6271
ia-enron	25	0.5	2	0.4643	0.7221	0.5651
fb-messages	100	0.5	1	0.5146	0.6072	0.5569
ia-realitycall	10	4	0.5	0.0329	0.0404	0.0362

Table 3 summarizes the results of MWU statistical tests with accompanying probabilities of superiority. $F_1(H)$ and $F_1(L)$ denote the average F_1 score for high NC-LID nodes (H) and low NC-LID nodes (L), respectively. U is the value of the MWU test statistic and the null hypothesis of the test is accepted if the p-value of U is higher than 0.05, which is also indicated by the ACC column in Table 3 ("no" means that the null hypothesis is rejected implying statistically significant differences between high and low NC-LID nodes w.r.t their F_1 scores).

Table 3. Comparison of F_1 scores of high NC-LID nodes (H) and low NC-LID nodes (L) using the Mann-Whitney U test.

Network	$F_1(H)$	$F_1(L)$	U	p	ACC	PS(H)	PS(L)
ia-realitycall	0.0863	0.0393	17873699	7.13E-43	no	0.1701	0.0817
college-msg	0.4761	0.5817	1485719.5	5.75E-30	no	0.3692	0.5804
email-eu	0.5998	0.6253	13980209.5	1.52E-13	no	0.4495	0.5308
radoslaw-email	0.4863	0.5729	127076.5	7.92E-10	no	0.3824	0.6022
ia-contacts	0.3482	0.5517	2279	3.13E-08	no	0.2272	0.7678
fb-forum	0.5869	0.6487	697028.5	4.81E-08	no	0.4032	0.5296
fb-messages	0.575	0.4836	1436845.5	1.46E-07	no	0.5194	0.4125
ia-hospital	0.4003	0.5499	908.5	0.0015	no	0.2882	0.7031
ia-primschool	0.7544	0.7629	26816	0.2448	yes	0.4688	0.5303
ia-enron	0.5256	0.5309	57862.5	0.7213	yes	0.4791	0.4949

The results of MWU tests reveal that there are statistically significant differences in F_1 scores between high NC-LID and low NC-LID nodes in 8 networks

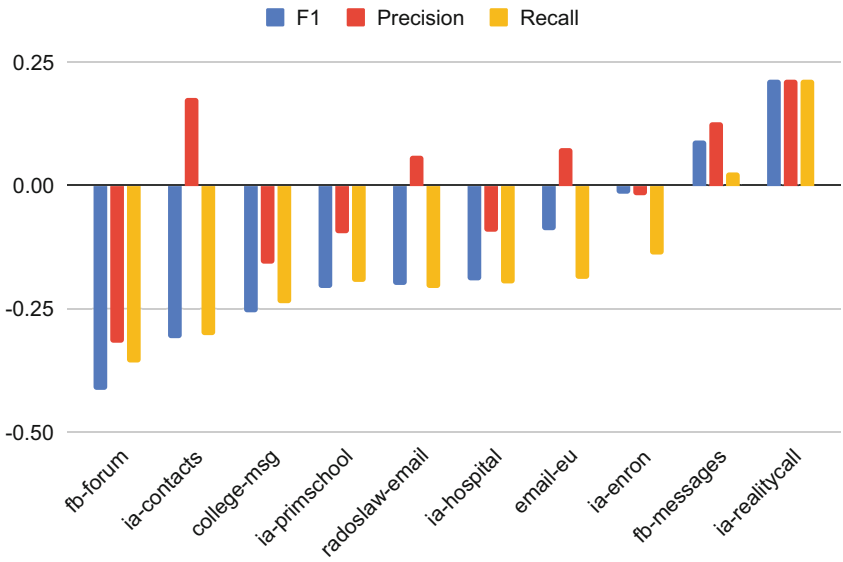


Fig. 3. Spearman correlations between NC-LID and graph embedding evaluation measures (F_1, precision and recall). The networks are sorted from the lowest to the highest correlation between NC-LID and F_1.

(the null hypothesis of the test is accepted for `ia-primschool` and `ia-enron`). Considering those 8 networks, in 6 of them we have that $F_1(H) < F_1(L)$ and $\mathrm{PS}(H) \ll \mathrm{PS}(L)$ implying that F_1 scores of high NC-LID nodes tend to be significantly lower than F_1 scores of low NC-LID nodes.

5 Conclusions and Future Work

In this paper we discussed how NC-LID, a LID measure originally designed for static graphs, can be adjusted for dynamic graphs, considering both their discrete-time and continuous-time representations. Then, following our previous works focused on static graphs, we proposed a methodology to examine the ability of NC-LID to point to weak parts of dynamic graph embeddings, i.e., to nodes whose embedding vectors do not tend to preserve their temporal neighborhoods. The obtained experimental results for the dynnode2vec dynamic graph embedding method on 10 real-world dynamic graphs indicate that NC-LID tend to negatively correlate to measures of intrinsic embedding quality obtained by the comparison of original graphs and graphs reconstructed from formed embeddings.

Our main finding suggests that NC-LID could be used to design LID-aware dynamic graph embedding algorithms. In our future work, inspired by our previous LID-elastic modifications of node2vec [13], we will examine similar modifications for dynnode2vec and other dynamic graph embedding methods based on

random walks. Since high NC-LID nodes tend to have low F_1 scores due to more complex intrinsic localities, their embedding vectors could be improved by personalizing hyper-parameters controlling random walk sampling strategies based on NC-LID values. For example, the number of random walks per node can vary so that a proportionally higher number of random walks is sampled for high NC-LID nodes, while keeping the total computational budget (the total number of random walk steps) constant. Second, hyper-parameters controlling random walk biases (transition probabilities) could be also adjusted according to NC-LID values in order to prevent prematurely leaving complex intrinsic localities.

Acknowledgments. This research is supported by the Science Fund of the Republic of Serbia, #7462, Graphs in Space and Time: Graph Embeddings for Machine Learning in Complex Dynamical Systems – TIGRA. The authors are grateful to anonymous reviewers for their comments and suggestions that helped improve the quality of the paper.

References

1. Barros, C.D.T., Mendonça, M.R.F., Vieira, A.B., Ziviani, A.: A survey on embedding dynamic graphs. ACM Comput. Surv. (CSUR) **55**(1) (2021). https://doi.org/10.1145/3483595
2. Grover, A., Leskovec, J.: Node2Vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, pp. 855–864. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939754
3. Holme, P., Saramäki, J.: Temporal networks. Phys. Rep. **519**(3), 97–125 (2012). https://doi.org/10.1016/j.physrep.2012.03.001
4. Houle, M.E.: Dimensionality, discriminability, density and distance distributions. In: 2013 IEEE 13th International Conference on Data Mining Workshops, pp. 468–473 (2013). https://doi.org/10.1109/ICDMW.2013.139
5. Khoshraftar, S., An, A.: A survey on graph representation learning methods. ACM Trans. Intell. Syst. Technol. **15**(1) (2024). https://doi.org/10.1145/3633518
6. Knežević, D., Babić, J., Savić, M., Radovanović, M.: Evaluation of LID-aware graph embedding methods for node clustering. In: Skopal, T., Falchi, F., Lokoč, J., Sapino, M.L., Bartolini, I., Patella, M. (eds.) Similarity Search and Applications, pp. 222–233. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-17849-8_18
7. Lancichinetti, A., Fortunato, S., Kertész, J.: Detecting the overlapping and hierarchical community structure in complex networks. New J. Phys. **11**(3), 033015 (2009). https://doi.org/10.1088/1367-2630/11/3/033015
8. Mahdavi, S., Khoshraftar, S., An, A.: dynnode2vec: scalable dynamic network embedding. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 3762–3765. IEEE Computer Society, Los Alamitos, CA, USA (2018). https://doi.org/10.1109/BigData.2018.8621910
9. Mann, H.B., Whitney, D.R.: On a test of whether one of two random variables is stochastically larger than the other. Ann. Math. Stat. **18**(1), 50–60 (1947). https://doi.org/10.1214/aoms/1177730491

10. Pan, R.K., Saramäki, J.: Path lengths, correlations, and centrality in temporal networks. Phys. Rev. E **84**, 016105 (2011). https://doi.org/10.1103/PhysRevE.84.016105
11. Savić, M., Ivanović, M., Jain, L.C.: Fundamentals of complex network analysis, pp. 17–56. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-91196-0_2
12. Savić, M., Kurbalija, V., Radovanović, M.: Local intrinsic dimensionality and graphs: towards LID-aware graph embedding algorithms. In: Reyes, N., et al. (eds.) Similarity Search and Applications, pp. 159–172. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-89657-7_13
13. Savić, M., Kurbalija, V., Radovanović, M.: Local intrinsic dimensionality measures for graphs, with applications to graph embeddings. Inf. Syst. **119**, 102272 (2023). https://doi.org/10.1016/j.is.2023.102272
14. Tomčić, A., Savić, M., Radovanović, M.: Hub-aware random walk graph embedding methods for classification. Stat. Anal. Data Min. ASA Data Sci. J. **17**(2), e11676 (2024). https://doi.org/10.1002/sam.11676

Urban Systems

An Application of Edge Betweenness Centrality in Bi-objective Optimization of Street Networks

Supharoek Chattanachot[1](✉), Frédéric Guinand[1], and Kittichai Lavangnananda[2]

[1] LITIS Laboratory, Le Havre Normandy University, Le Havre, France
supharoek.chattanachot@etu.univ-lehavre.fr , frederic.guinand@univ-lehavre.fr

[2] Faculty of Science, Technology and Medicine, Luxembourg University, Luxembourg, Luxembourg
kittichai.lavangnananda@ext.uni.lu

Abstract. Due to climate change, earth temperatures are continuously increasing globally and this trend is even more pronounced in cities, where heat islands phenomena pose a danger to public health. At the city level, transforming part of the street surface area into green spaces may be an effective approach for mitigating this climate effect. The main objective of this work is to design new methods for reducing street surfaces under specific distances and strong connectivity constraints. The algorithms developed build Pareto Font for this bi-objective problem. They operate on directed weighted graphs derived from OpenStreetMap. The main contributions are two new algorithms, both based on betweenness centrality, the Iterative Edge Betweenness Centrality (IEBC) and Self-adaptive Iterative Edge Betweenness Centrality (S-IEBC). Ten French cities were selected for their various characteristics. In validation, the performances of the two algorithms proposed together with previous methods are compared. S-IEBC is proven superior to all previous methods.

Keywords: Bi-objective optimization · Betweenness centrality · Green cities · OpenStreetMap · Reduction of street lanes · Strongly connected weighted graph · Urban network

1 Introduction

According to the conclusions of the work of the Intergovernmental Panel on Climate Change (IPCC) [1], *Human activities, principally through emissions of greenhouse gases, have unequivocally caused global warming, with the global surface temperature reaching* 1.1°C *above 1850–1900 in 2011–2020.* This trend continues, as revealed by the last figures provided by the Copernicus Climate Change Service (C3S), which uses six different datasets [2]. Over the last 60

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 389–401, 2025.
https://doi.org/10.1007/978-3-031-82439-5_32

months, they reported a global increase of 1.3°C. The situation is even more pronounced when considering only land areas: an increase of more than 1.8°C, and in some regions, such as Europe, where the increase is approximately 2.3°C. It has also been shown that in cities, and especially downtown, temperatures are higher than in their rural surroundings. This phenomenon is known as the urban heat island (UHI) effect, and the difference in temperature between downtown areas and rural surroundings can be around 4°C [2]. To reduce this heat stress, some strategies involve increasing green areas within cities. However, these efforts are hindrance challenged by the observed global trend of increasing the number of streets in cities worldwide [3]. In response, various initiatives have been proposed to mitigate these effects, such as the EU's Zero-Emission Vehicle initiative, which aims to reduce emissions by 55% for cars and 50% for trucks. Other projects focus on reducing the surface area occupied by streets for other purposes, including the creation of green spaces, for instance, the successful Superblock project implemented in Barcelona [4].A similar concept, known as the'road diet' has been implemented in the US for years [5].It involves repurposing street lanes for other uses, which improves safety and quality of life and provides benefits for the built environment. The work presented in this paper follows the same idea, focusing on reducing the surface area occupied by streets for greening purposes. We formulate this as a bi-objective optimization problem, focusing on reducing the number of lanes and limiting the increase in point-to-point distances within the city. The model considered is a digraph, and the main contributions of this work are two new algorithms based on the betweenness centrality measure for building Pareto fronts.

The paper is organised as follows. Section 2 describes the model, the objectives, and the constraints and also reports related works. The central idea of the proposed algorithms is presented in Sect. 3. Section 4 provides results from the experiments. Conclusion and some perspectives are discussed in Sect. 5.

2 Models and Related Works

2.1 Model, Objectives and Constraints

Graphs are popular models for representing street networks [6]. In this study, a directed weighted graph model is adopted to represent a street network where an edge represents a lane on the street and specific points such as junctions, intersections, and roundabouts are represented by nodes. For simplicity, multiple lanes are modelled as a single arc. Thus, when two arcs are adjacent to the same vertices, they are orientated in opposite directions. Such a scenario is referred to as 'symmetric or opposite arcs' in this paper.

The main objective is to reduce the number of lanes in the street network. However, removing lanes may inevitably increase point-to-point distances within the city. Additionally, careful consideration must be given when lane removal is selected to preserve strong connectivity property. Therefore, the problem can be formulated as a bi-objective optimization problem subject to the constraint of preserving the strong connectivity property of the directed graph. The first

objective that should be minimized is the number of preserved arcs that represent preserved lanes in a street network. The second objective that should also be minimized is the All-Pair-Shortest-Path (APSP) value, which represents the sum of the distances between all distinct couples of nodes in the network.

Apart from strong connectivity, some additional constraints are considered. Adding new lanes/arcs and reversing the direction of the current lanes/arcs are prohibited operations. If two nodes are connected by only one arc, this arc cannot be removed (i.e. removing it is very likely to increase APSP value). Finally, the number of nodes cannot be altered (i.e. addition or subtraction) by the algorithm, every node in the original graph must also be present in the solution. Therefore, for a given $G = (V, A)$, the original directed graph, all proposed solutions (modified graphs) and denoted $G' = (V', A')$ are such that $V' = V$ and $A' \subset A$.

2.2 Previous Works

The work in this paper is part of the Urban Transportation Network Design Problems (UTNDP). Using the terminology and classification of Farahani [7], it is positioned at the tactical level of road network design problems (RNDP), and more specifically belongs to the Discrete Network Design Problem (DNDP). However, if making some streets one-way is considered in some works, their general objectives are not exactly the same. Indeed, the main objective of this work is to minimize both the number of arcs in a directed graph and the APSP value. To date, only the works in [8] and [9] have the same objectives.

In these works, the authors consider the same bi-objective optimization problem and propose some algorithms for building a Pareto front. However, there are two main differences between these works and the work in this paper. At the model level, a preliminary work in [8] used only synthetic data, whereas real city maps were considered in [9]. In the latter, however, road lengths were not taken into account, so a very short street was treated as equivalent to a much longer main street in terms of the second objective. The second and most important difference lies in the algorithmic approach to determining arc removal. Their proposed methods focus on selecting cycles from which arcs can be removed, whereas this work takes a different approach by considering the importance of each arc in the street network based on its betweenness centrality measure, as will be described in the next section. Additionally, Wiedemann et al. recently proposed an interesting approach for balancing bike and car networks [10]. Their work aims to reallocate lanes from car traffic to bike lanes. The two objectives they consider are increasing car travel time and reducing perceived bike travel time. A linear programming formulation of the problem is proposed to construct a Pareto front. Apart from these recent works, preserving strong connectivity exists in several previous works. The earliest this kind is likely to be the work of Robbins in 1939, who proposed and proved the following theorem: “A graph is orientable if and only if it remains connected after the removal of any arc” [11]. However, arc reversals were allowed and the same can be stated for several

works which followed. This is now commonly referred to as a street reconfiguration problem. However, none of these works considers the possibility of removing arcs with the objective of minimizing the number of lanes in the network. In the next section, the general process for building a solution is elaborated along with the two newly proposed algorithms.

3 Proposed Methodology

The method for removing lanes while maintaining the graph's strong connectivity is based on processing cycles in the directed graph. Consider a set of nodes that belong to the same cycle $(u_0, u_1, u_2...u_k, u_0)$. If along this cycle, there exist opposite arcs (u_j, u_{j-1}), then any of these arcs can be removed while preserving the strong connectivity of the entire graph. The algorithms proposed in [9] are also based on this property. The general process for building solutions was composed of three phases: (i) cycle detection, (ii) cycle selection, and (iii) arc removal. The detection and choice of cycles were made regardless of their possible impacts on the computation of the second objective, the APSP.

In this work, the same three phases are kept, but unlike in the previous works, opposite arcs are not systematically removed. Removal selection depends on the relative importance of each arc regarding the shortest paths between nodes. The importance of each arc is determined by its betweenness centrality (BC) [12]. For a given arc, the BC value is the number of shortest paths passing through this arc for all pairs of graph vertices. Since the APSP is our second objective, using the BC value is the most straightforward criterion to limit the increase of this objective. Removing higher BC arcs will indicate a greater impact on this objective. Thus, preserving arcs with high BC values will be a key consideration in our algorithms presented in the following sections. For the computation of the betweenness centrality measure, we consider the fraction of shortest paths between all pairs of vertices that pass through a given arc: $\text{EBC}_a = \sum_{s,t \in V} \frac{SP(s,t|a)}{SP(s,t)}$ where, $SP(s,t)$ is the total number of shortest paths, and $SP(s,t|a)$ is the total number of shortest paths passing through the arc a. Then, during the resolution process, after the detection of a cycle, the removal of opposite arcs depends on their BC values, but a reference BC value is needed for making the decision, we call it the Edge Betweenness Centrality Preservation Threshold (EBC-PT), Its determination and its use are described in the following section.

3.1 Edge Betweenness Centrality Preservation Threshold: EBC-PT

The Edge Betweenness Centrality Preservation Threshold (EBC-PT) is introduced as a heuristic to help the algorithm decide which arc can be removed. The EBC-PT value, ranging from 0 to 1, represents the range of the EBC values of arcs that will be conserved and not allowed to be removed from the network. The value must be defined at the beginning of the process and computed to be a ratio of the highest EBC value among all arcs in the network.

Suppose EBC-PT is defined at 0.2, then the arcs which EBC values belong to the top 20 percent of the highest EBC values of the graph will not be allowed to be removed from the network. We can thus compute a threshold above which arcs cannot be removed. More formally, given $G = (V, A)$, such that $\forall a \in A, \text{EBC}_{min} \leq \text{EBC}_a \leq \text{EBC}_{max}$, the threshold, denoted by h, is equal to:

$$h = \text{EBC}_{max} - ((\text{EBC}_{max} - \text{EBC}_{min}) * \text{EBC-PT}) \tag{1}$$

Suppose $\text{EBC}_{min} = 0.1$ and $\text{EBC}_{max} = 0.8$ and a EBC-PT equals to 0.2, then $h = 0.8 - ((0.8 - 0.1) * 0.2) = 0.66$. This means all arcs with an EBC value greater than 0.66 will be conserved in the network while some arcs with a lower EBC value might be removed. At the two extremes, when EBC-PT is null, then any arc might be removed from the network, while when EBC-PT is equal to one, no arc can be removed from the network. Generally, the greater the EBC-PT value, the lower the possibility of removing arcs from the network because many of the arcs with a high EBC value are conserved. Of course, the removal of any arc in the network may change the EBC value of other arcs, so in the algorithms described in the next sections, one operation consists in updating the threshold. The method is described by Algorithm 1.

Algorithm 1: Threshold update method

Input: Street Network G, EBC-PT value

```
thresholdUpdate():
    Compute EBC (G)
    determine EBC_max and EBC_min
    h ← EBC_max − ((EBC_max − EBC_min) ∗ EBC-PT)
```

Then, when a new graph has been obtained by removing some arcs, the quality of the solution for both objectives and its non-dominance within the Pareto Front has to be evaluated. The method is described by Algorithm 2.

Algorithm 2: Pareto front update method

Input: Street Network G

```
paretoFrontUpdate():
    V1 ← number of symmetric arcs in G
    V2 ← all pairs shortest paths value in G
    S ← (V1, V2)
    if S is non-dominated then
        add S to the Pareto front
    end
```

The following sections present two algorithms based on this threshold: Iterative Edge Betweenness Centrality (IEBC) and Self-adaptive Iterative Edge

Betweenness Centrality (SIEBC). The general process is the same for both algorithms. The difference lies in managing the threshold, which is key to deciding which arc to remove.

3.2 Iterative Edge Betweenness Centrality (IEBC)

The resolution process is iterative. Initially, two variables must be defined: the pre-processed street network G, a strongly connected directed graph, and the EBC-PT value. As discussed earlier, the algorithm searches for cycles to remove arcs while preserving the strong connectivity property. For each new cycle detected, opposite arcs are examined, and depending on their EBC value, they are either removed or retained. If some arcs are removed, the iteration is considered successful. Conversely, when no arc can be removed, either because there is no opposite arc or their EBC values are greater than h, the iteration is unsuccessful. The process is described in Algorithm 3.

Algorithm 3: Cycle detection and Opposite Arcs removal method

```
Input: Street Network G, threshold h
Output: the success boolean
cycleDetectionAndOppositeArcsRemoval():
    success ← false
    C ← DetectCycle()
    Ā ← opposite arcs of C
    for each a ∈ Ā do
        if EBC_a < h then
            remove(a) from G
            success ← true
        end
    end
    return success
```

The algorithm allows a maximum number of unsuccessful iterations, denoted by p, before terminating the process. This provides an opportunity to find a better solution. It is possible that different cycle choices can help the algorithm escape from a plateau state. Note that after each arc removal, the EBC values of the arcs may change, including EBC_{max} and EBC_{min} values.

Therefore, after a successful iteration, these two values and h must be recomputed. This is done by the method *thresholdUpdate()*. The solution obtained from the new graph is then evaluated, using the *paretoFrontUpdate()* method, in relation to the current Pareto front, and added to it if it is non-dominated. The entire process is described by Algorithm 4.

One drawback of this method is the need for an prior choice of a constant EBC-PT value, from which the threshold h is computed. As illustrated on Fig. 1, the selection of a specific EBC-PT value affects both the quality of the second

Algorithm 4: Iterative Edge Betweenness Centrality

```
Input: Street Network G, EBC-PT value, p
Main():
    thresholdUpdate()
    nbOfUnsuccessfulIterations ← 0
    while nbOfUnsuccessfulIterations < p do
        success ← cycleDetectionAndArcRemoval()
        if success then
            paretoFrontUpdate()
            thresholdUpdate()
            nbOfUnsuccessfulIterations ← O
        else
            nbOfUnsuccessfulIterations++
        end
    end
```

objective (APSP), and the algorithm's ability to remove many arcs from the network, which is the first objective. The next algorithm addresses this issue by allowing dynamic adaptation of the EBC-PT value during the resolution process.

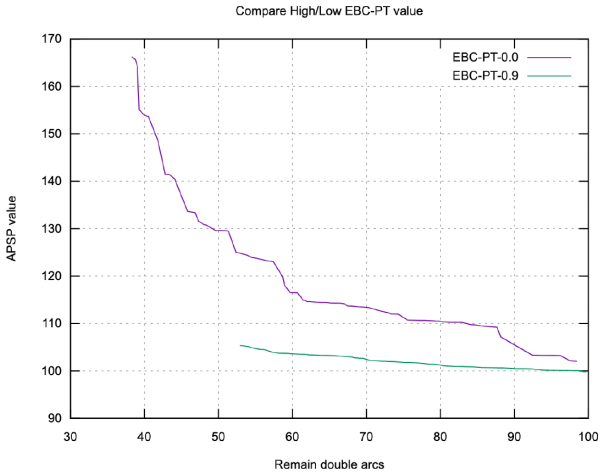


Fig. 1. Compare High/Low EBC-PT value, Honfleur, France

3.3 Self-adaptive Iterative Edge Betweenness Centrality (S-IEBC)

The Self-adaptive Iterative with Edge Betweenness Centrality (S-IEBC) makes the EBC-PT value self-adjusting during the process. As illustrated by Fig. 1, a

value of EBC-PT close to 1 favors the quality of objective 2 (APSP), while a value closer to 0 enables the algorithm to remove more arcs, thus improving the first objective.

Algorithm 5: SIEBC

```
Input: Street Network G, p
Main():
    EBC-TP_current ← EBC-PT_max
    thresholdUpdate()
    nbOfUnsuccessfulIterations ← 0
    while EBC-PT_current > EBC-PT_min do
        success ← cycleDetectionAndArcRemoval()
        if success then
            paretoFrontUpdate();
            thresholdUpdate();
            nbOfUnsuccessfulIterations ← O
        else
            nbOfUnsuccessfulIterations++
            if nbOfUnsuccessfulIterations = p then
                nbOfUnsuccessfulIterations ← 0
                EBC-PT_current ← EBC-PT_current − EBC-PT_step
            end
        end
    end
```

Thus, the self-adjustment of the EBC-PT value involves incrementally decreasing this value throughout the process. To address this, after several unsuccessful iterations, the EBC-PT value is decreased, resulting in a larger h, and consequently, a larger number of arcs becoming eligible for removal. In summary, in the early stages, the algorithm removes a few arcs while limiting the increase of the APSP value (objective two). In later stages, the maximum benefit is obtained for objective one, by eliminating arcs with EBC values larger than in the early stages. Algorithm 5 describes the method. In the algorithm, EBC-PT$_{max}$ denotes the intial value of EBC-PT and EBC-PT$_{min}$ represents the lowest considered value. For a given EBC-PT value (denoted by EBC-PT$_{current}$ in the algorithm), when the maximum number of unsuccessful iterations is reached, EBC-PT current value is decreased by the amount denoted as EBC-PT$_{step}$.

4 Experiments and Analysis

4.1 From Real Maps to Graphs

The proposed algorithms are evaluated using ten real-world street networks. Cities of varying sizes, ranging from small to medium, within France, were

selected for this study. The characteristics of these selected cities are presented in Table 1. The datasets are sourced from OpenStreetMap (OSM) [13], which provides the necessary features for analysis. The boundaries of the selected cities are provided by OSM. Note that these boundaries may intersect with some outgoing streets, which could have an impact, especially for small cities. The networks were restricted to streets that can be used by vehicles. Pedestrian paths, stairs, and other street types not intended for cars were filtered out, along with dead-end streets and self-loop streets, which are not meaningful for our study. The data preparation process converts OSM data into graph models based on GraphStream [14].

Table 1. Graph characteristics of a selection of French cities. Obj 1 is the number of symmetric arcs and Obj 2 corresponds to the All Pairs Shortest Paths (APSP) values that are expressed in thousands of km.

City	Honfleur	Bernay	Vichy	Périgueux	Troyes	Auxerre	Valence	Amiens	S.Etienne	Reims
# Arcs	1,025	1,066	1,730	1,889	2,736	2,895	3,906	5,224	5,550	7,198
# Nodes	482	449	772	818	1,231	1,273	1,841	2,478	2,414	3,186
Obj. 1	397	436	547	682	936	1,060	1,246	1,618	1,984	2,418
Obj. 2	692	478	909	1,403	3,637	4,695	11,742	22,812	23,131	42,788

The generated graphs are directed, weighted graphs, with arcs weighted by the actual street lengths obtained from OSM data. Several lanes going in the same direction are represented by only one arc. Note that this simplification has no impact on the second objective (APSP value) but penalizes the first objective, since removing two parallel lanes instead of one would better improve this objective. When the graph is generated, its strong connectivity is verified using Kosaraju algorithm. [15].

4.2 Results and Discussion

This section concentrates on the results obtained from the proposed algorithms. The objectives of this work are to minimize both the number of symmetric arcs in the graph (first objective), and the APSP value (second objective) of the solutions. As these two objectives are contradictory, the non-dominated solutions are represented as a Pareto front. Three analyses are conducted. The first focuses on the impact of EBC-PT value on the quality of the solutions produced by the two algorithms and enable us to conclude the superiority of the S-IEBC algorithm compared to IEBC. The second focuses on the results obtained by S-IEBC for ten different French cities. The last analysis involves a comparison of the performance of algorithms previously designed to address this problem [9] with the current ones.

Impact of EBC-PT. The proposed algorithms are driven by the h threshold value, which is derived from the EBC-PT value. A high EBC-PT value results in a small h value, restricting arc removal to those with low EBC values. Removing arcs with low EBC values is less likely to negatively impact the APSP objective. Conversely, when EBC-PT value is low, many arcs may be candidates for removal, which contributes to minimizing the number of symmetric arcs, the first objective.

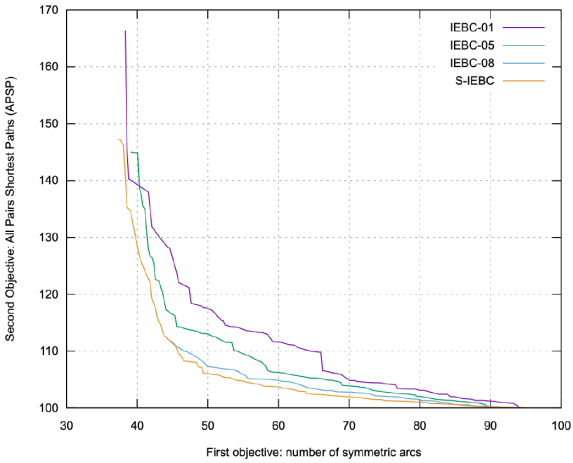


Fig. 2. IEBC and S-IEBC algorithms run for Honfleur map. Three constant values of EBC-PT are tested for IEBC: 0.1, 0.5 and 0.8.

The two algorithms, IEBC and S-IEBC are executed on the Honfleur map and the results are reported on Fig. 2. As illustrated by the Pareto front obtained for each configuration, S-IEBC clearly outperforms all parameterized version of IEBC, regardless of the EBC-PT value chosen. Further experiments conducted on ten other cities confirm the same results. It should also be noted that for any value of the first objective (number of remaining symmetric arcs), the higher the EBC-PT value the better the second objective. Thus it seems that solutions produced by IEBC-high dominate solutions produced by IEBC-low, but at the cost of a truncated Pareto front, since IEBC-high is unable to find solutions with a low percentage of remaining symmetric arcs, so unable to minimize the first objective.

Impact of the City Map. In this second analysis, we run S-IEBC on ten different cities and the obtained Pareto fronts are represented in Fig. 3.

The Pareto front associated with Honfleur is very different from those obtained for all the other cities.This can be explained by the particular characteristics of this small historical city. The administrative boundaries of the city intersect many outgoing streets leading to neighbouring cities. As a result, many

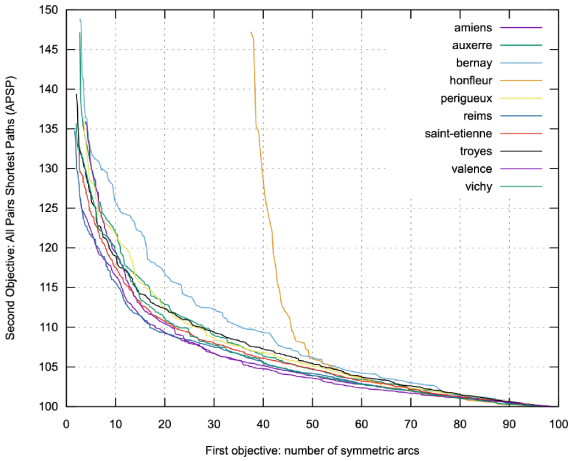


Fig. 3. Comparison of S-IEBC among ten cities (for each instance the algorithm was run ten times). Note the particular shape of the Pareto front for Honfleur. The small size of the city, along with its strong connections to neighboring cities, may explain the limitation in reducing the number of symmetric arcs.

vertices (not dead-ends) are connected by symmetric arcs to only one neighbour node. Therefore, removing these symmetric arcs would lead to losing the strong connectivity property. The second unexpected result comes from the very similar shape of the Pareto Front for all the other cities. Regardless of the city, removing half of the symmetric arcs increases the APSP by around 5%. Our hypothesis is that the selected French cities have similar topological characteristics, but this should be further investigated by considering more cities and also cities from other countries and continents.

Comparison with Previous Algorithms. Three algorithms for addressing the problem were proposed in [9]. The different methods were dedicated to the detection of cycles.The Iterative Depth-First Search (I-DFS) method is based on a simple DFS procedure for detecting cycles.It is the simplest one and serves as a basis for the next two.The Two-Compatible Cycles Method (2CCM) detects cycles with common arcs, while The Two-Compatible Cycles with Relaxation Method (2CCRM) is a trade-off between I-DFS and 2CCM.The results reported in [9] show the complementarity of the three methods for building Pareto fronts.However, none of them can efficiently build a Pareto front that covers a wide range of values for the first objective.

The comparison of the performances of these previous three methods with the two new ones is illustrated in Fig. 4. The city of Bernay is chosen for the experiments but the same results hold for other cities.It can be clearly seen that, without any ambiguity,the S-IEBC algorithm outperforms all others, including the methods previously introduced in [9].

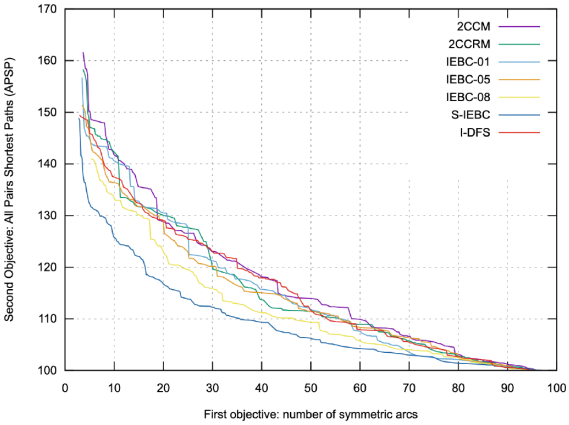


Fig. 4. Comparison of arcs removal method on Bernay, France

5 Conclusion and Perspectives

To mitigate the effects of climate change, specifically heat islands in cities, creating green spaces districts may be part of the solution. Identifying these spaces presents a real challenge, and repurposing lanes dedicated to cars is worth investigating. In this work, we propose two algorithms, based on the betweenness centrality, to achieve this goal. The experiments show that the Self-Adaptive Iterative Edge Betweenness Centrality (S-IEBC) algorithm outperforms all previously known methods. From an urban network perspective, the results also show that more than 50% of two-way, two-lane streets can be reduced to one-way, one-lane streets with a very limited increase in the sum of all distances within the city (about 5%). The results also reveal that further investigations into cities' characteristics are necessary to determine if the same Pareto front shape is commonly shared among cities with specific attributes.

The approaches proposed in this work focus on performances in terms of two objectives: reducing the number of lanes in the network and maintaining all pairs shortest paths (APSP) values. In practice, however, the proposed methods may be subject to criticism. Indeed, neither the impact on traffic nor constraints such as decongesting downtown, reversal direction of streets, nor many other parameters, have been taken into account. Traffic conditions will be incorporated in the next phase of this research. We plan to account for traffic in various ways, either by maintaining the same flows or by studying the maximum traffic capacity within the newly defined urban network. Nevertheless, the work presented in this paper ought to be a good starting point and pave the way to a more comprehensive approach. This is an ongoing research into the attempt to improve urban network environments by repurposing the street lanes.

Acknowledgement. The authors are grateful to the Royal Thai Government for providing a Ph.D. scholarship to Mr. S. Chattanachot and gratefully acknowledge FSTM, the University of Luxembourg's support, and the ANR-22-CE22-0002-ANR Magnetics project France.

References

1. IPCC: Climate Change 2023: Synthesis Report, Summary for Policymakers (2023)
2. C3S: Copernicus climate change service (2024). https://climate.copernicus.eu/climate-indicators/temperaturehttps://climate.copernicus.eu/demonstrating-heat-stress-european-cities
3. Barrington-Leigh, C., Millard-Ball, A.: Global trends toward urban street-network sprawl. Proc. Nat. Acad. Sci. **117**(4), 1941–1950 (2020)
4. Mueller, N., et al.: Changing the urban design of cities for health: the superblock model. Environ. Int. **134**, 105132 (2020)
5. Knapp, K., et al.: Road diet informational guide, Technical report, United States, Federal Highway Administration, Office of Safety (2014)
6. Boeing, G.: OSMnx: new methods for acquiring, constructing, analyzing, and visualizing complex street networks. Comput. Environ. Urban Syst. **65**, 126–139 (2017)
7. Farahani, R.Z., Miandoabchi, E., Szeto, W.Y., Rashidi, H.: A review of urban transportation network design problems. Eur. J. Oper. Res. **229**(2), 281–302 (2013)
8. Chattanachott, S., Guinand, F., Lavagnananda, K.: How to reduce streets-network sprawl? arXiv preprint (2023)
9. Chattanachott, S., Guinand, F., Lavagnananda, K.: Bi-objective optimization for reducing urban network sprawl. In: 2024 16th International Conference on Knowledge and Smart Technology (KST), pp. 265–270. IEEE (2024)
10. Wiedemann, N., Nobel, C., Martin, H., Ballo, L., Raubal, M.: Bike network planning in limited urban space. arXiv preprint arXiv:2405.01770 (2024)
11. Robbins, H.E.: A theorem on graphs, with an application to a problem of traffic control. Am. Math. Mon. **46**(5), 281–283 (1939)
12. Freeman, L.: A set of measures of centrality based on betweenness. Sociometry **40**(1), 35–41 (1977)
13. Open Street Map: Open street map (2014). https://www.openstreetmap.org
14. Dutot, A., Guinand, F., Olivier, D., Pigné, Y.: Graphstream: a tool for bridging the gap between complex systems and dynamic graphs. In: Proceedings of Emergent Properties in Natural and Artificial Complex Systems (ECCS'2007), pp. 63–72 (2007)
15. Sharir, M.: A strong-connectivity algorithm and its applications in data flow analysis. Comput. Math. Appl. **7**(1), 67–72 (1981)

Multiple Route Search Method on Road Network Using Genetic Algorithm with Virus Infection

Haruto Aoyagi(✉) and Takayasu Fushimi

Tokyo University of Technology, Tokyo, Japan
{c0b2100287,fushimity}@edu.teu.ac.jp

Abstract. Due to the need for driving routes that emphasize scenery, it is an important research topic to explore various routes while passing through predetermined spots. In this study, we propose a method that expresses the spots that should be passed through by a virus and generates a route that satisfies the constraints on passing through the spots by infection. In addition, by performing tournament selection and elite selection in each cluster obtained through clustering, generational change while maintaining diverse routes is realized. Through evaluation experiments targeting two urban networks in Japan, we confirm the effectiveness of the proposed method in terms of route diversity and constraint sufficiency by comparing it with a regular genetic algorithm without virus infection and clustering.

Keywords: route search · road network · genetic algorithm

1 Introduction

Research on route search has been conducted in a wide range of fields. In particular, studies on optimal route and multiple route exploration have found applications in logistics, traffic systems, robot navigation, travel planning, and many other areas. Representative methods for the most basic problem, the shortest path problem, include Dijkstra's algorithm [3], which efficiently finds the shortest path in non-negative weighted graphs, the A* algorithm [7], which improves search efficiency by using heuristic evaluation functions, and the Bellman-Ford algorithm [1,5], which can handle graphs with negative weights.

In the field of network route exploration, research particularly focused on urban roads and traffic networks has been conducted [2,10,13]. For example, some methods utilize real-time traffic information to avoid road congestion, while others derive optimal routes by simulating the movement of vehicles and pedestrians. These methods serve as important tools for realizing efficient movement while adapting to congestion and traffic volume fluctuations. In multiple route exploration, there is a demand not only for a single optimal route but also for providing diverse candidate routes. The K shortest path algorithm by Yen is

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 402–414, 2025.
https://doi.org/10.1007/978-3-031-82439-5_33

frequently used [14]. Thus, research on route exploration has been ongoing for many years, and recently, there has been active exploration of multiple routes and routes emphasizing scenic views [6]. However, methods that emphasize the similarity or diversity between the explored routes are limited, and many approaches tend to present relatively similar routes.

In this study, we propose a method for exploring sub-optimal multiple routes on road networks based on a genetic algorithm (GA) with an emphasis on route diversity. The GA is an optimization method that mimics the process of biological evolution and is a type of metaheuristic algorithm used particularly to find approximate solutions to complex problems [8]. The basic principle is to approach the optimal solution by mimicking evolutionary processes such as natural selection, crossover (combination), and mutation (random changes). The proposed method begins by uniformly and randomly sampling intersection nodes in the road network and connecting them to form initial individuals. This allows the length of each individual's genes to be fixed to the same size and enables efficient processing even for large-scale road networks. In addition, during the generational transition, individuals are clustered and selection is performed within each cluster to maintain route diversity. Then, by performing crossover all individuals across clusters, it is possible to produce diverse individuals that are independent of the initial cluster. Furthermore, routes that satisfy user-desired conditions (such as scenic views) are introduced as virus individuals, and each route individual can become infected by nearby viruses with a certain probability, allowing routes that meet the conditions to be obtained. Evaluation experiments using actual large-scale road networks were conducted to verify the effectiveness of the proposed method from the perspectives of route length optimality (short routes), route diversity, and the fulfillment of conditions such as scenic views.

2 Related Work

This section reviews related work on multiple-routes search and route searches using genetic algorithms.

2.1 Multiple-Routes Search

Research on multiple route exploration involves the development of technologies and methods aimed at finding several routes simultaneously. This includes not only providing optimal paths but also suggesting diverse routes according to user needs. The K shortest paths problem seeks to find K shortest paths from a starting point to a destination in addition to a single shortest path. This allows users to choose from multiple routes. To address this problem, Yen's K shortest paths algorithm is widely used [14]. Yen's algorithm efficiently explores alternate paths similar to the shortest path found. Furthermore, Eppstein's K shortest paths algorithm offers a more efficient method to find K shortest paths and is applicable to large-scale graphs [4].

Nishimura et al. proposed a method to search multiple Cyclic routes for strolling [12]. In their method, a cyclic path is explored by setting up a quadrangle with four vertices including the starting point, and connecting the vertices so that the path passes through as many POIs as possible. By setting multiple quadrangles that satisfy the path length constraint, multiple paths are explored with minimal overlap. It is similar to our study in that it searches for a variety of routes by minimizing overlap among multiple routes. On the other hand, it differs from ours in that it can only deal with the maximization problem of passing through as many POIs as possible. Our method searches for a quasi-optimal route that satisfies the point constraints of stopping places such as gas stations and convenience stores, the line constraints of arterial roads, and the area constraints of scenery such as nature and townscapes unlike their method.

2.2 Route Searches Using Genetic Algorithms

Maruyama et al. proposed a personal navigation system (PNS) for efficient navigation among multiple destinations [11]. In the PNS, tourists can specify multiple destinations, their desired arrival and stay times, and priorities. The system calculates and navigates a route that includes some of the destinations that satisfy the tourist's requirements. To this end, the authors developed a route search algorithm using a genetic algorithm, which can retains multiple candidate solutions during search, and give approximate solutions. The PNS consists of two steps: 1) searches the shortest path between each pair of destinations; 2) determines the visiting order of the destinations. The main purpose of PNS is to use multiple routes obtained through GA search to respond to route changes during a trip, so it retains individual routes with very high similarities between multiple routes. On the other hand, this research differs in that it takes into account the need to retain a variety of routes, i.e. routes with low similarities.

Kanoh and Nakamura used a genetic algorithm incorporating virus infection to address the problem of finding a route that stops at multiple non-specific points according to the driver's requirements [9]. Here, non-specific points refer to stopover points specified by category, such as gas stations, banks, and convenience stores, without specifying a specific store. In their study, partial routes that include these stopover points are kept as virus individuals. Virus infection occurs between two points that are common to the candidate route from the start point to the destination (normal individuals) and the virus individuals, and exchanges intersection genes. Through intersection and infection, a quasi-optimal route that is the easiest to drive while passing through the stopover points is determined. Experiments using actual road maps showed that this infection-based mechanism is an effective way to solve the problem. Following this method, we also represent partial routes including not only spots but roads and areas using virus individuals, and search for routes that satisfy constraints through infection according to the distance between individuals and viruses.

3 Proposed Method

In this study, we treat the road structure of the target city as directed network $G = (\mathcal{V}, \mathcal{E})$, where a node $v \in \mathcal{V}$ and an edge $e \in \mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ correspond to an intersection and a street between intersections, respectively. Each edge $e = (u, v)$ is weighted with distance $w(e) = d(u, v)$. Here, $d(u, v)$ is the straight-line distance calculated based on the coordinates of nodes u, v, and is calculated as $d(u, v) = \sqrt{(y_u - y_v)^2 + (x_u - x_v)^2}$, where y_v, x_v are the latitude and longitude of node v.

3.1 Problem Setting

In this study, we address the problem of searching multiple suboptimal routes from a starting node $s \in \mathcal{V}$ to a destination node $t \in \mathcal{V}$ on a road network G and that satisfy constraints. Each route is represented by a sequence of n nodes on the route $\mathbf{r} = [v_1, v_2, \ldots, v_n]$ and the total distance of the route is $w(\mathbf{r}) = \sum_{m=1}^{n-1} w((v_m, v_{m+1}))$, where $v_1 = s$ and $v_n = t$. Suboptimal is defined as the ratio $w(\mathbf{r})/w(\mathbf{r}^*_{s,t})$ of the shortest path $\mathbf{r}^*_{s,t}$ from s to t to the total distance $w(\mathbf{r}^*_{s,t})$ is less than ε. The constraint we deal with in this study is to pass through spot nodes on the road network. In order to broaden the user's options, it is necessary to search for a variety of routes with low similarity, rather than searching for multiple similar routes.

3.2 Overview of Proposed Method

In this study, we propose a method for searching a set of routes using a genetic algorithm that introduces clustering to maintain the diversity of routes and viral infection to satisfy the constraint that passes through spots. In the proposed algorithm, each route is treated as an individual. The process flow is as follows:

1. Initial individual generation: Generate I individuals;
2. Clustering: Divide I individuals into K groups;
3. Selection: Perform elite selection and tournament selection for each cluster;
4. Crossover: Perform crossover for the individuals selected by tournament selection;
5. Mutation: Reinitialize the immunity of each individual with a certain probability;
6. Virus infection: Infect each individual with a nearby virus with a certain probability;

Clustering and tournament selection for each cluster maintain the diversity of routes. Mutation and virus infection are used to satisfy the constraints. Linearity is satisfied by sorting the landmark genes at crossover. The process from STEP2 to 6 is repeated T times, a specified number of generations. The following sections describe each step in detail.

3.3 Initialization

First, we set a parameter λ_1 that adjusts the distance from the starting node s and the destination node t, and define a landmark probability for each node $v \in \mathcal{V}$ according to the distance: $\mathrm{P}_{\mathrm{ldm}}(v; \lambda_1) = \exp\left(-\lambda_1\{d(s,v) + d(v,t) - d(s,t)\}\right)$. The closer a node is to the shortest path between the starting node s and the destination node t, the higher the probability of it being selected. Based on this probability, we sample L intersection nodes and use them as landmarks $\mathcal{U} \subset \mathcal{V}$.

In the initial individual generation, we randomly select $\ell - 2$ landmarks, excluding the starting point and destination, from the length of the individual's chromosome ℓ, and sort and rearrange them by the distance from the starting point to the destination. That is, an individual i is represented by an array (chromosome) of ℓ landmarks: $\mathbf{h}_i = [u_1, \ldots, u_\ell]$, where $u_1 = s$ and $u_\ell = t$. By connecting adjacent landmarks with the shortest path using Dijkstra's algorithm, we obtain a route $\mathbf{r}_i = [v_1, \ldots, v_{n_i}]$ from the starting point s to the destination t, where $v_1 = s$ and $v_{n_i} = t$.

Next, for each individual, we define the immunity that controls the susceptibility to virus infection. The higher the immunity, the less susceptible the individual is to virus infection. As a constraint for route search, sometimes it is sufficient to pass through a spot once, and other times it is necessary to pass through a spot multiple times. To control for these, we set the initial immunity as $\beta_i \leftarrow \mathrm{Beta}(a, b)$ using the beta distribution.

This process generates I initial populations $\mathcal{I}^{(0)} = \{i_1, \ldots, i_I\}$.

3.4 Clustering

To select individuals while maintaining diversity, all I individuals are divided into K clusters. By performing tournament selection for individuals for each cluster that groups similar routes, even if the individual has high fitness, there is a high probability that the individual with other similar routes will be eliminated. Conversely, even if the individual has low fitness, there is a high probability that it will survive if there are few similar individuals. As a result, it is possible to maintain more diverse routes than with a normal GA. In addition, in the next generation and beyond, individuals can cross over across clusters to generate more diverse routes.

The similarity between each individual is measured using Dynamic Time Warping (DTW), and the cluster number c_i for each individual is determined by clustering using the k-means method.

3.5 Evaluation and Selection

When performing tournament selection and elite selection, the fitness of each individual is needed. Let $\mathcal{O}$ be the set of O virus individuals. The fitness f_i of an individual i is quantified by mixing the total distance $w(\mathbf{r}_i)$ of the individual to be minimized and the constraint satisfaction with a ratio parameter π:

$$f_i = \pi \frac{w(\mathbf{r}^*_{s,t})}{w(\mathbf{r}_i)} + (1-\pi) \sum_{o \in \mathcal{O}} \delta(\mathbf{g}_o \cap \mathbf{h}_i \neq \varnothing).$$

Here, $\delta(\text{cond})$ is a function that returns 1 when the condition cond is satisfied, and 0 otherwise, which indicates whether there is an overlap between the virus o and the route i, i.e., whether the route passes through a spot represented by a virus. The individual with high fitness is considered to be an excellent individual.

Selection is to decide which individuals to live in the next generation. For each cluster $\mathcal{I}_k = \{i \in \mathcal{I}; c_i = k\}$, we perform elite preserving selection and tournament selection based on fitness f_i.

3.6 Crossover

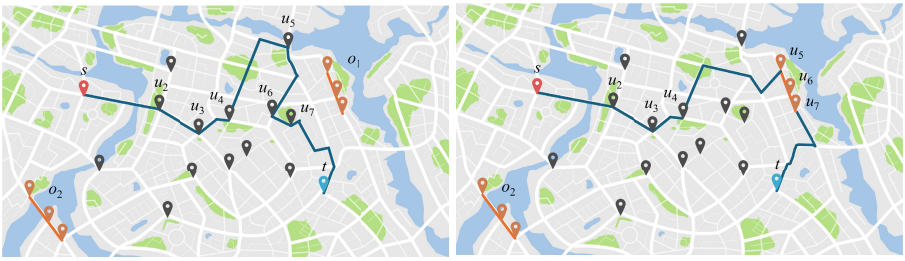
Crossover is a genetic operation that generates new individuals that inherit the properties of their parents with a certain probability. Specifically, the individuals selected by tournament selection are treated as targets regardless of clusters, and two individuals are randomly selected with a certain probability γ to be parents. Parts of the genes are cut from the chromosomes of the two parents and connected to create chromosomes of children. There are single-point and multi-point crossovers, and the proposed method uses multi-point crossover. Since landmark genes are exchanged, zigzag routes are likely to be generated, which may delay convergence. Therefore, in the proposed method, landmark genes are exchanged using a projection method to maintain the linearity of the route.

3.7 Mutation

Mutation is an operation that changes some of the genes of an individual born from crossover with a certain probability. In the proposed method, genetic manipulation is performed by viral infection, so mutation is used to change immunity to viral infection. In the proposed method, as explained in the next section, each individual's immunity increases each time it is infected with a virus, decreasing the probability of infection with other viruses.

3.8 Viral Infection

Viral infection is an operation that replaces part of an individual's genes with viral genes according to the infection probability and immunity. In the proposed method, a route that satisfies the constraints by viral infection, i.e., a route that passes through a specified spot, is generated. A virus individual is constructed by the spot node q that must be passed through and the intersection nodes sampled from within a certain range of q. A value of λ_2 is set to adjust the distance from the spot node q and the intersection node v, and a virus probability according to the distance is defined for each intersection node: $\mathrm{P}_{\mathrm{vrs}}(v; \lambda_2) = \exp(-\lambda_2 d(q, v))$. In accordance with this probability, intersection nodes near the spot are sampled for $l-1$, where $l = |\mathbf{g}_o|$ is the length of the chromosome $\mathbf{g}_o$ of the virus o.



(a) Virus individual (b) Gene replacement in viral infections

Fig. 1. Toy example of proposed method

As in the case of generating the initial individuals, the viral genes are sorted and rearranged by the distance from the starting point to the destination, and adjacent genes are connected by the shortest path using the Dijkstra algorithm. Figure 1(a) shows an example in which there are two viruses, o_1 and o_2, with chromosome length $l = 3$.

We define the infection probability according to the distance between each route and the virus present everywhere in the target area. In the proposed method, the closer a virus o is to a route individual i, the higher the infection probability. Specifically, we set the value of λ_3 to adjust the distance between the spot $q \in \mathbf{g}_o$ of virus o and the gene $u \in \mathbf{h}_i$ of route i, and define the infection probability as follows: $\mathrm{P}_{\mathrm{inf}}(o, i; \lambda_3) = \exp\left(-\lambda_3 \min_{u \in \mathbf{h}_i} d(q, u)\right)$. If the infection probability $\mathrm{P}_{\mathrm{inf}}(o, i; \lambda_3)$ is higher than the immunity θ_i, the virus o infects the individual i. Figure 1(b) shows an example of infection by virus o_1, where the individual's genes $u_5, u_6, and u_7$ are replaced by the virus' genes. Here note that, if an individual already has a spot in its chromosome, it will not be reinfected with a virus that contains that spot. This trial is made for all viruses.

Each time an individual is infected with a virus, its immunity increases. The immunity θ_i of an individual i against a virus is determined by the following function with the number of infections j as a variable:

$$\theta_i(j; \alpha, \beta) = \frac{1}{(1.5 - \beta)^{-1} + \exp(-\alpha j)} - 0.5 + \beta$$

Here, α and β represent the immunity increase rate and initial immunity, and the initial immunity is set as a random number following the beta distribution $\mathrm{Beta}(a, b)$ when the initial individual is generated or when a mutation occurs. The higher the immunity increase rate α, the more rapidly the immunity increases. If the immunity increase rate α is small, the immunity increases more slowly. By increasing immunity with the immunity increase rate α, it is possible to control the susceptibility to infection by a virus. The number of infections j is not passed on to offspring individuals.

4 Experiments

In evaluation experiments, we use real data to demonstrate the effectiveness of the proposed method in terms of route diversity through clustering and constraint satisfaction through virus infection.

4.1 Dataset and Parameters

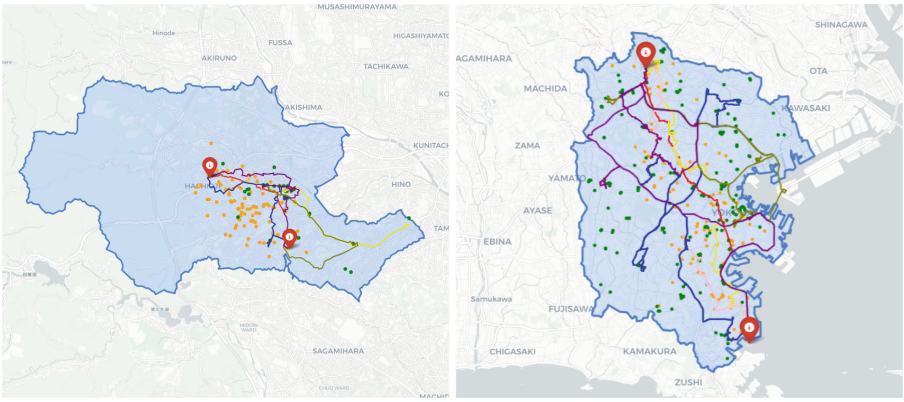
Table 1. Basic statistics of road network

	Hachioji	Yokohama
Number of nodes $\lvert\mathcal{V}\rvert$	17,192	90,753
Number of edges $\lvert\mathcal{E}\rvert$	47,022	246,752
Average degree	5.47	5.44
Average shortest path length (graph distance)	58.96	70.17
Diameter (graph distance)	160	182
Average shortest path length (actual distance) [m]	6,568.89	12,786.06
Diameter (actual distance) [m]	31,985.27	40,479.35

Road and spot data for two cities, Hachioji and Yokohama, are collected from OpenStreetMap[1]. Figure 2 shows the road network of Hachioji and Yokohama, the shortest route from the starting point to the destination, and landmarks and virus spots. In the Hachioji data, the starting point is Starbucks Motohachioji (35.66490639001417, 139.28782643798056), the destination is Tokyo University of Technology (35.62620801031389, 139.3405264254352), and the shortest distance is 8.27 km (time required: 13.14 min). In the Yokohama data, the starting point is Misuzugaoka Park (35.56337381740233, 139.54019119659824), the destination is Hakkeijima Sea Paradise (35.33643171592319, 139.64521916775564), and the shortest distance is 30.76 km (travel time: 46.31 min). Table 1 shows basic statistics for the road network in Hachioji and Yokohama. In our experiments, we used a bank as the spot to be passed through. The number of banks is 33 in Hachioji and 229 in Yokohama, respectively.

The method proposed in this study involves multiple parameters. We change the values of some of the parameters to confirm the effectiveness of the proposed method. The default settings for each parameter are as follows: Number of individuals $I = 100$, number of generations $T = 50$, neighborhood control parameters $\lambda_1 = \lambda_2 = \lambda_3 = 1.0$, number of landmarks (including start and end points) $L = 100$, length of individual chromosomes $\ell_i = 20$, number of clusters $K = 5$, tournament size $J = 10$, number of tournament selections $\kappa_{\mathrm{trn}} = 2$, number of elite selections $\kappa_{\mathrm{elt}} = 1$, crossover probability $\gamma = 0.5$, mutation

[1] https://www.openstreetmap.org.

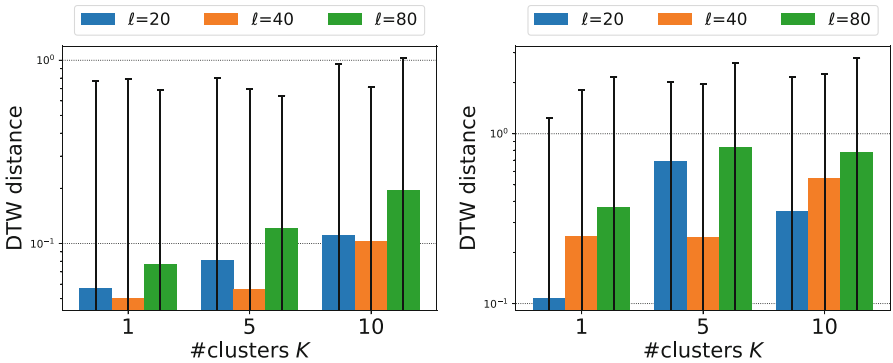


(a) Hachioji (b) Yokohama

Fig. 2. Road network: Red pins are starting and ending nodes, red line is shortest path between them, and orange and green points are landmarks and virus spot. The non-red lines are the elite individuals in the five clusters obtained by the proposed method.

probability $\mu = 0.1$, beta distribution parameters $(a, b) = (2, 5)$, length of virus chromosomes $l_o = 5$, radius of virus [m]: 500, immunity increase rate $\alpha_i = 1.0$, the fitness mixture ratio $\pi = 0.5$.

4.2 Evaluation of Route Diversity



(a) Hachioji (b) Yokohama

Fig. 3. DTW distance w.r.t the number of clusters

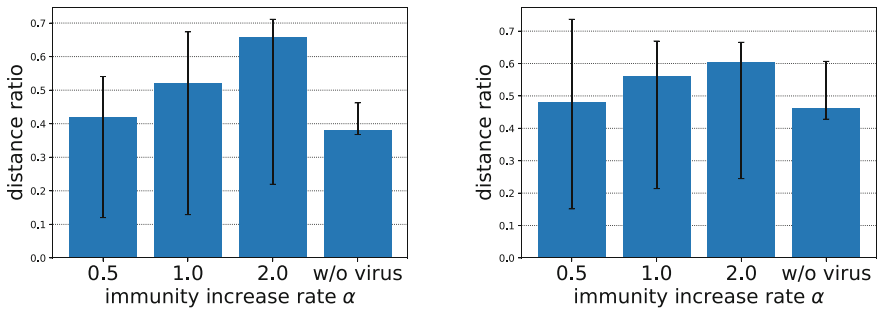
In this experiment, we change the number of clusters K to evaluate whether the proposed method that introduces clustering can search for diverse routes.

The parameters in this experiment were set as follows: Number of clusters $K \in \{1, 5, 10\}$, chromosome length of each individual $\ell_i \in \{20, 40, 80\}$, mixture ratio $\pi \in \{0.5, 1.0\}$. The number of clusters $K = 1$ is equivalent to no clustering. The purpose of clustering is to maintain diversity. Therefore, we use the DTW distance of each individual as an evaluation index. A larger average DTW distance means that diversity is better maintained.

In Fig. 3, we plot the minimum, average, and maximum of DTWs $[\mathrm{dtw}(i,j)]_{(i,j)\in\mathcal{I}\times\mathcal{I}}$ between all pairs of individuals by error bars. The horizontal axis shows the number of clusters K, and the vertical axis shows the DTW distance on a logarithmic scale. As shown in Fig. 3(a), the average DTW distance is the largest and routes are most diverse when the number of clusters is $K = 10$ and the chromosome length is $\ell_i = 80$. It can also be said that for any chromosome length, the more clusters there are, the more diverse routes can be explored. In Fig. 3(b), we can see that, with some exceptions, the more clusters there are, the more diverse routes with larger distances between routes can be explored.

These results show that it is difficult to search for diverse routes without clustering, and that the clustering introduced in the proposed method makes it possible to search for diverse routes.

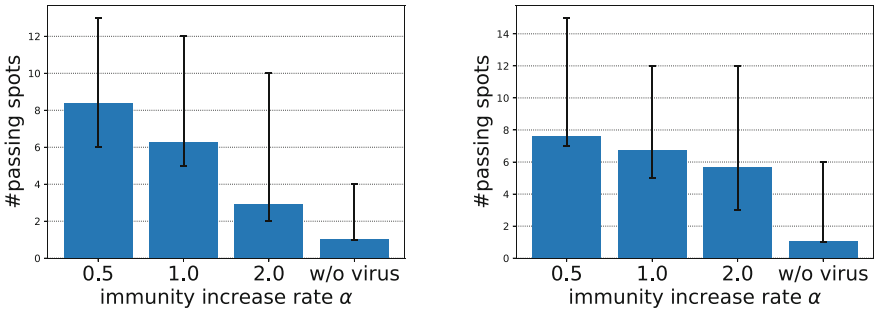
4.3 Evaluation of Constraint Satisfaction



(a) Hachioji (b) Yokohama

Fig. 4. Distance ratio w.r.t the immunity increase rate

In this experiment, we change the immunity increase rate to evaluate whether the proposed method that introduces virus infection can search for a route that satisfies the constraints. To confirm the effectiveness of virus infection, we compare the results with those without virus infection. The parameters in this experiment were set as follows: Immunity increase rate $\alpha_i \in \{0.5, 1.0, 2.0\}$. The purpose of virus infection is to search for a route that satisfies the constraints. For this reason, we use the ratio of the individual route distance to the shortest route length $w(\mathbf{r}^*_{s,t})/w(\mathbf{r}_i)$ and the number of spots included in the route



(a) Hachioji (b) Yokohama

Fig. 5. Number of passing spots w.r.t the immunity increase rate

$\sum_{o \in \mathcal{O}} \delta(\mathbf{g}_o \cap \mathbf{h}_i \neq \varnothing)$ as evaluation indices. The closer the ratio to the shortest route length is to 1, the higher the optimality, which means that the destination can be reached without making too many detours. In addition, we confirm the influence of virus infection by checking the number of spots passed through.

Figure 4 plots the immunity increase rate α_i on the horizontal axis and the distance ratio on the vertical axis. Figure 4 shows that for all data, the average distance ratio was closest to 1 when the immunity increase rate $\alpha_i = 2.0$. In a normal GA without virus infection, the distance ratio and spot passing constraints are imposed only by fitness, which is thought to have resulted in an low distance ratio.

Figure 5 plots the immunity increase rate α_i on the horizontal axis and the number of passed spots on the vertical axis. Figure 5 shows that for all data, the immunity increase rate $\alpha_i = 0.5$ was the most frequent. This is because the higher the immunity increase rate, the higher the immunity each time the individual is infected, which has the effect of suppressing the number of infections. In the absence of virus infection, the number of passed spots is smaller than in the proposed method that introduces virus infection.

As can be seen from these two results, by introducing viral infection it is possible to search for a route that satisfies the constraints. Without viral infection even if a spot-passing constraint was introduced to the fitness, the route would be longer and would not necessarily pass through the spots. We also found that the susceptibility to viral infection can be controlled by setting the immunity: setting the immunity increase rate low increased the number of passing spots and reduced the ratio to the shortest distance, while setting the immunity increase rate high reduced the number of passing spots and increased the ratio to the shortest distance.

5 Conclusion

In this study, we proposed a method to search for multiple routes that take into account scenic routes and stopovers, rather than simply moving to a destination. Our method searches for routes that satisfy constraints by introducing virus infection into a genetic algorithm. In addition, by performing clustering, generations were changed while maintaining diversity. Evaluation experiments using actual road networks confirmed that our method is superior to other methods in terms of diverse route search and constraint satisfaction.

In this experiment, the effectiveness of only point constraints was confirmed for two cities. In the future, we plan to develop a route search that is city-independent and supports line constraints and area constraints. As the target area becomes larger, the processing time increases, so we need to consider an algorithm to speed up the process. In addition, the method requires the setting of a variety of parameters. We need to conduct experiments under various conditions to confirm the dependency of each parameter on the data size and find appropriate values for each parameter.

Acknowledgement. The second author is grateful for the financial support from JSPS Grant-in-Aid for Scientific Research (C) (No.22K12279).

References

1. Bellman, R.: On a routing problem. Q. Appl. Math. **16**(1), 87–90 (1958)
2. Chen, B., Liu, H.: A dynamic traffic assignment model considering real-time traffic information. Transp. Res. Part C Emerg. Technol. **15**(5), 380–400 (2007)
3. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numer. Math. **1**(1), 269–271 (1959)
4. Eppstein, D.: Finding the k shortest paths. In: Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 154–163. SIAM, San Francisco, CA, USA (1998)
5. Ford, L.R., Fulkerson, D.R.: Network flow theory. RAND Corporation Report (1956)
6. Fushimi, T., Otaki, H.: A method for presenting multiple routes considering distance and landscape probability for automotive navigation systems. In: 2023 IEEE International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), pp. 249–255. IEEE Computer Society, Los Alamitos, CA, USA (2023)
7. Hart, P.E., Nilsson, N.J., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. In: IEEE Transactions on Systems Science and Cybernetics, vol. 4, pp. 100–107. IEEE (1968)
8. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Reprinted by MIT Press, 1992, Ann Arbor, MI (1975)
9. Kanoh, H., Nakamura, N.: Route guidance with unspecified staging posts using genetic algorithm for car navigation systems. In: 2000 IEEE Intelligent Transportation Systems, pp. 119–124 (2000). https://doi.org/10.1109/ITSC.2000.881028
10. Li, Z.C., Huang, Z., Zhou, Z.: Urban traffic flow prediction based on a hybrid approach of neural network and traffic flow theory. Transp. Res. Part B Methodol. **57**, 159–175 (2013)

11. Maruyama, A., Shibata, N., Murata, Y., Yasumoto, K., Ito, M.: A personal tourism navigation system to support traveling multiple destinations with time restrictions. In: 18th International Conference on Advanced Information Networking and Applications, 2004, vol. 2, pp. 18–21. AINA 2004 (2004)
12. Nishimura, T., Ishikawa, K., Takayama, T., Yanagisawa, M., Togawa, N.: A multiple cyclic-route generation method for strolling based on point-of-interests. In: 2018 IEEE 8th International Conference on Consumer Electronics - Berlin (ICCE-Berlin), pp. 1–6 (2018). https://doi.org/10.1109/ICCE-Berlin.2018.8576185
13. Wang, J.L., Chou, L.Y.: Real-time optimal route planning based on the simulation of traffic flows. J. Eastern Asia Soc. Transp. Stud. **6**, 2573–2588 (2005)
14. Yen, J.Y.: Finding the k shortest loopless paths in a network. Manage. Sci. **17**(11), 712–716 (1971)

Optimising Urban Electric Vehicle Charging Infrastructure: A Spatial Modelling Approach for Fleet Operators with Case Studies in Mumbai and Delhi

Sedar Olmez(✉) and Sachio Kobayashi

Fujitsu Research of Europe, 9 Albert St, Slough, Berkshire SL1 2BE, UK
sedar.olmez@fujitsu.com

Abstract. The global rise in electric vehicle (EV) adoption highlights the need for expanded charging infrastructure. A significant gap exists between affluent nations, such as G20 members, and others in developing this infrastructure. While wealthier nations utilize advanced spatial modelling, others often rely on ad-hoc methods lacking quantitative rigor. This paper proposes a spatial modelling approach using open-source and private data tailored for fleet operators. Our method optimally identifies locations and capacities for new urban charging stations via a novel software pipeline. Fleet operators can leverage this to select warehouse locations, enhance infrastructure, and perform scenario analyses in an agent-based model. A case study in Mumbai and Delhi demonstrates substantial improvements: a 70% reduction in charger waiting times, a 20% decrease in time to reach a charger, and a 70% reduction in charging duration, among other key performance indicators.

Keywords: Electric Vehicle Charging Infrastructure · Spatial Model · Agent-Based Model

1 Introduction

The suitable location identification and construction of new charging infrastructure in cities has become an increasingly important topic over the years with the increased uptake of electric vehicles across the world sometimes incentivised by governments [3,5] to combat the rise in green house gasses from internal combustion engine vehicles (15% of which is from the transportation sector [2]). There are several perspectives in achieving optimised charging infrastructure planning, some important conditions are:

- Minimisation of infrastructure costs, thus, identifying a configuration where the number of stations is as small as possible [4,14,18].
- Profiteering, increased usage of chargers to increase income for charging infrastructure providers, identifying a configuration where usage will be highest [1].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 415–424, 2025.
https://doi.org/10.1007/978-3-031-82439-5_34

- Minimisation of costs to user, identifying configurations where the charge waiting time, distance to charger is smallest [8,9].
- Maximised usage, identifying configurations where the number of EV slots is highest [16,19,22,23].

Given the aforementioned list of explored conditions in the modelling literature, an evident problem arises where these conditions are explored independently. We believe some of these conditions should be incorporated into a single model and bear weight on the locations of these charging infrastructures. To overcome this issue, we propose a novel spatial modelling pipeline which utilises several data sources including that of a private fleet operator with several stages consisting of underlying algorithms to identify optimal charging infrastructure locations and size.

To test the methodology, we identified two cities, namely Delhi and Mumbai in India as the fleet operator that provided their data, operated in these locations. The aggregated results showed a ∼70% drop in the waiting time for a charger and average duration of the charging activity, a ∼20% drop in the average time it takes to reach a charger and ∼70% drop in the total time spent charging based on twenty-four new charging infrastructure locations based in the aforementioned cities. The experiments were initially conducted with the current state of charging infrastructure based on available open-source data in Delhi and Mumbai and compared to the same simulations with the new charging infrastructure adopted.

The proposed methodology is spatially agnostic, therefore, any geography can be utilised so long as demand data is present with population density (usually from a Census). Furthermore, charging infrastructure is not the only domain in which the approach can be utilised in, for example, new shopping facilities can be identified, local governments can plan the locations of new social housing projects and so on.

The following sections consist of a literature review 2, the methodology 3, results are disseminated with the setup of the experiments in 4 and finally the findings and conclusions are discussed in 5.

2 Literature Review

According to Metais et al. [15], the placement of charging infrastructure and the speed of charging are important considerations in the planning of charging infrastructure. However, some studies have concentrated solely on location without factoring in capacity or the number of charging slots [10,22]. To develop a more comprehensive infrastructure plan, we assert that population density and fleet demand are crucial factors that should inform the identification of both location and capacity.

The identification of a charging infrastructure's size, i.e., the land parcel, can be made by estimating the anticipated number of electric vehicles (EVs) that will require each station [23]. We align with this approach, utilising empirical data on fleet activities, particularly demand, and population density to guide this

decision. Our methodology not only incorporates demand from existing EVs but also considers the potential future adoption of EVs over Internal Combustion Engine Vehicles (ICEVs), allowing our infrastructure plan to potentially accommodate the impact of future policy changes aimed at boosting EV sales in Delhi and Mumbai.

Our findings highlight the significance of incorporating charging infrastructure capacity into the planning process to yield an optimal outcome. Failure to consider this aspect could result in suboptimal outcomes [15], leading to significant redevelopment costs for developers and diminished utility of these stations. Capacity adjustments can sometimes be achieved by constructing new stations in close proximity, thereby mitigating the impact of malfunctioning stations [15]. By leveraging a clustering algorithm, our newly identified charging infrastructure are spatially proximate to one another.

Another critical consideration is the time required for charging. While extended charging durations may not always be inconvenient, there are instances where quicker charging is imperative. Hence, methodologies such as power sizing, which designate appropriate charging capacities across stations [21,24], become crucial. In our model, we opted for level two charging, which offers greater efficiency compared to level one charging primarily used in household charging facilities.

Two well-established approaches for determining the location of electric vehicle charging infrastructures are the node-based and path-based methodologies. The former involves exhaustive testing of various configurations randomly placed across space to identify nodes with the highest demand for charger placement. However, this approach is NP-hard and becomes increasingly complex with larger node sizes [17]. In contrast, the path-based approach utilises flow-capturing methods, positioning charging infrastructure along paths with the highest traffic flow [12]. Our approach aligns with the latter, utilising real-world fleet operation data from Mumbai and Delhi for individual origin-destination trips. We extend beyond road constraints, acknowledging that certain locations may be more efficient if developed and later integrated into a road network. Additionally, we incorporate population density into our methodology to weigh location and capacity, we also utilise agent-based modelling to simulate the impact of our new charging infrastructure locations.

Another prevalent methodology employed by researchers is the p-median approach [7], which selects locations weighted by transportation costs (i.e., distance), favouring locations with the lowest distance [6,11,13,20].

While the aforementioned methodologies have contributed significantly to identifying optimal charging infrastructure locations, this article proposes a novel approach. Our methodology integrates several crucial factors, including population, demand, capacity, and location, through a distinct set of steps in our software pipeline (see Fig. 1).

3 Methodology

This section describes the methodology adopted, specifically detailing the software pipeline and the models used.

Our proposed methodology utilises several techniques, including an agent-based model that simulates demand dynamics based on input fleet data for delivery services across the chosen city. The individual-level data generated from this simulation is subsequently fed into a k-Means clustering algorithm. This process helps to consolidate population density and demand patterns into a unified dataset, which serves as input for algorithms designed to identify optimal charging locations. The approach for determining suitable slot capacities is based on a linear equation with population density. For a comprehensive understanding, refer to the high-level methodology depicted in Fig. 1.

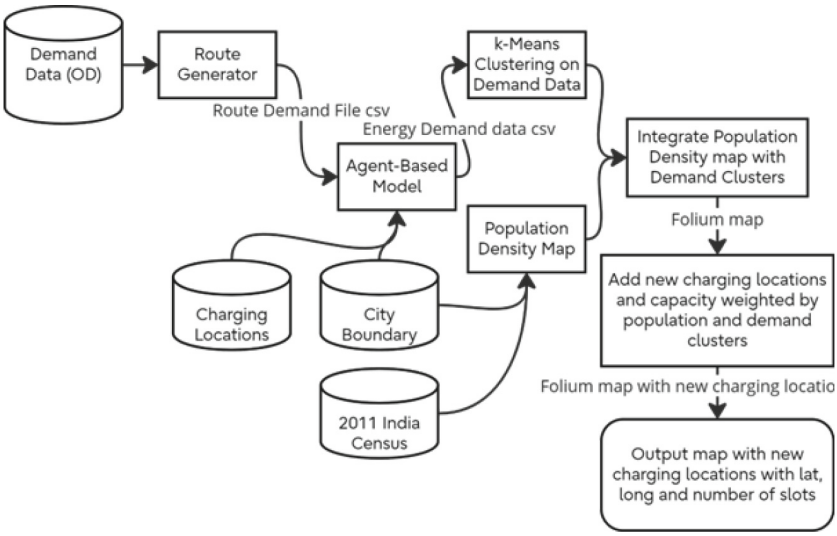


Fig. 1. High-level flow diagram of the proposed methodology.

The algorithm for determining the placement of charging infrastructure, depicted in the lower right corner of Fig. 1, starts by setting up a structure called *new_charging_locations*. This structure includes fields for latitude, longitude, slots, and centroid, designed to hold the details of newly identified charging spots. The core procedure involves processing each point from a pre-selected group of centroids. For every centroid, the algorithm notes its latitude and longitude. It then computes how far each point in the provided dataset is from this centroid, appending these distances to the data. Points within a certain distance from the centroid are filtered and organised by population density in descending order. From this ordered list, the algorithm selects the top four points. The number of slots for each charging location is decided based on the population density

of these points. Finally, the algorithm records the new charging location details, including its geographic coordinates, slot number, and centroid reference, into *new_charging_locations* without duplicating existing entries.

In the following section, we describe the results and experiments conducted in both Delhi and Mumbai, India as the input data was captured in these cities.

4 Results

This section presents experimental findings from four distinct experiment conditions. Our experiments consist of the following steps:

1. Assessment of the current state of charging infrastructure in Delhi and Mumbai.
2. Evaluation of scenarios involving both the existing and proposed charging infrastructure in Delhi and Mumbai.

The input data is based on individual trips by 250 vehicles, conducting fleet operations for a one month period. The datasets also include; charging locations and city boundaries, the charging locations consisted of the latitude and longitude of the open-source data for electric vehicle charging infrastructure in both cities. The city boundaries and census data were from the same source, the population density was the main focus, this includes the population at the local level in the order of a hundred thousand. In Table 1, we present the details of these data.

Once the pipeline from Fig. 1 concludes, the outputs can be visualised in an interactive map as shown in Fig. 2 and Fig. 3 for Delhi and Mumbai respectively, here we see the location of new charging infrastructure and the size. Additionally, a subset of Key Performance Indicators (KPIs) comparing different experimental conditions are summarised in Table 2. These KPIs, derived from the ABM simulations, provide insights into the efficiency of the newly identified charging locations in optimising overall fleet operations. The columns labelled 'pre' and 'post' denote conditions before and after the integration of new charging infrastructure into fleet operations, respectively.

The following list describes each of the KPIs provided in Table 2:

- Average charging time per task: the mean duration, in minutes, required for a vehicle to complete its charging process.
- Average in-route time to charge per task: the average time, in minutes, taken by a vehicle to reach charging infrastructure.
- Total charging time: the cumulative duration, in minutes, spent by vehicles in the charging process.
- Total waiting time to charge: the overall duration, in minutes, vehicles spent queuing for access to charging infrastructure.

Table 1. Pipeline input data for fleet demand and charging infrastructure

Column	Data type	Description
Number of vehicles	Integer	The fleet size.
Vehicle model	String	The model of the vehicle used.
Power train	String	Electric, Petrol etc.
Range (KM)	Float	The range the vehicle can do in distance from a single charge.
Min-charge threshold	Float	The percentage of charge at which a vehicle decides to go to a charging station.
Battery capacity	Float	The battery capacity of the vehicle.
Pickup	Float	The latitude and longitude of the pick-up location.
Drop-off	Float	The latitude and longitude of the drop-off location.
Latitude and longitude of re-fuel station	Float	The locations of the existing petrol/charging stations.
Type	String	The type of refuel station, electricity, petrol or both.
Fill rate	Float	The kilowatt capacity of the amount of energy that can be drawn from the station.
Slots	Integer	The number of charging slots and/or parking slots available at the station.

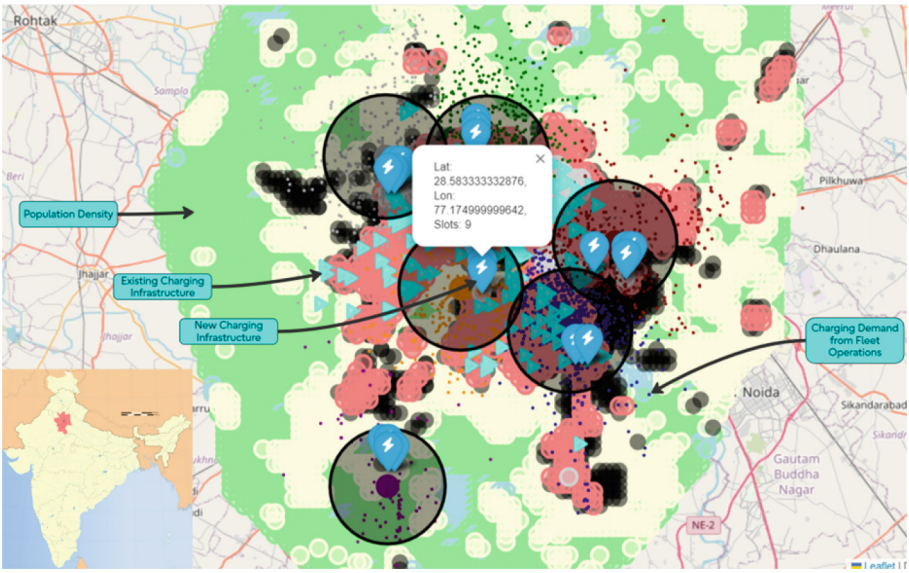


Fig. 2. New charging infrastructure locations in Delhi.

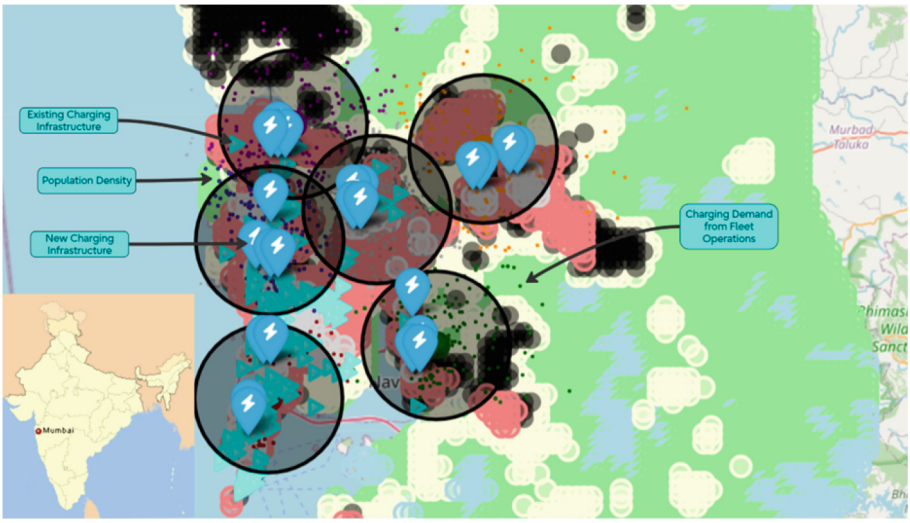


Fig. 3. New charging infrastructure locations in Mumbai.

Table 2. Aggregated KPIs related to EVs for both Mumbai and Delhi before and after new charging infrastructure.

KPI	Delhi (pre)	Delhi (post)	Mumbai (pre)	Mumbai (post)
Average charging time per Task [m]	167.19	48.66	261.88	78.03
Average in-route time to charge per task [m]	7.46	5.99	8.50	7.14
Total charging time [m]	715,092.33	214,097.00	393,090.33	114,474.16
Total waiting time to charge [m]	217.16	42.83	87	0.16

Our experimental findings demonstrate consistent improvements across both cities, evident in reductions in charging time, waiting durations for charging, and the time taken to reach charging infrastructure. These results, coupled with our modelling approach, lay the groundwork for future development. In upcoming iterations, we plan to incorporate additional KPIs such as cost savings and carbon emissions, enabling policymakers to forecast the potential impact of infrastructure interventions on these critical metrics.

5 Conclusion and Discussion

In conclusion, this paper has presented a novel spatial modelling approach for the optimal placement and capacity of electric vehicle charging infrastructure in urban areas. The methodology leverages data resources available to fleet operators, including population density and fleet demand, to identify the most efficient

locations for new charging infrastructure. The approach was tested in two major Indian cities, Mumbai and Delhi, and demonstrated significant improvements in key performance indicators, including a reduction of approximately 70% in waiting time for a charger, a decrease of around 20% in the average time required to reach a charging infrastructure, and a drop of approximately 70% in the average charging duration.

The results of this study have significant implications for the development of EV infrastructure, particularly in countries where such infrastructure is currently lacking. By employing advanced spatial modelling techniques, these countries can ensure that their charging infrastructure is optimally placed to meet the needs of EV users, thereby encouraging the adoption of EVs and contributing to the reduction of greenhouse gas emissions.

However, this study is not without its limitations. The methodology relies on the availability of detailed data on fleet operations and population density, which may not be available in all contexts. Furthermore, the study focused on the placement of charging infrastructure in urban areas, and further research is needed to determine whether the same approach can be applied in rural or less densely populated areas.

In the future, this technology will be extended by incorporating CO2_kg savings calculations and cost of fuel compared to electricity using application programming interfaces (APIs). This will show the user how much money was saved through the new EV infrastructure and how fleet operations can be tailored to the locations in an optimal way. This will not only enhance the efficiency of fleet operations but also contribute to the broader goals of reducing carbon emissions and promoting sustainable transportation.

Overall, this study represents a significant step forward in the development of EV infrastructure and provides a strong foundation for future research in this area.

Acknowledgements. We would like to thank the World Business Council for Sustainable Development (WBCSD) for their continued support throughout the project and Gentari for sharing data and insights on fleet operations.

References

1. Cai, H., Jia, X., Chiu, A.S., Hu, X., Xu, M.: Siting public electric vehicle charging stations in Beijing using big-data informed travel patterns of the taxi fleet. Transp. Res. Part D Transp. Environ. **33**, 39–46 (2014)
2. Cook, J., et al.: Quantifying the consensus on anthropogenic global warming in the scientific literature. Environ. Res. Lett. **8**(2), 024024 (2013)

3. Diamond, D.: The impact of government incentives for hybrid-electric vehicles: evidence from US states. Energy Policy **37**(3), 972–983 (2009)
4. Dong, X., Mu, Y., Jia, H., Wu, J., Yu, X.: Planning of fast EV charging stations on a round freeway. IEEE Trans. Sustain. Energy **7**(4), 1452–1461 (2016)
5. Gong, S., Ardeshiri, A., Rashidi, T.H.: Impact of government incentives on the market penetration of electric vehicles in Australia. Transp. Res. Part D Transp. Environ. **83**, 102353 (2020)
6. Gorbunova, A., Anisimov, I.: Optimizing the location of urban charging stations for electric vehicles: case study of the city of Tyumen, Russian federation. Urban Transp. XXVI **200**, 15 (2020)
7. Hakimi, S.L.: Optimum locations of switching centers and the absolute centers and medians of a graph. Oper. Res. **12**(3), 450–459 (1964)
8. Hanabusa, H., Horiguchi, R.: A study of the analytical method for the location planning of charging stations for electric vehicles. In: Knowledge-Based and Intelligent Information and Engineering Systems: 15th International Conference, KES 2011, Kaiserslautern, Germany, September 12-14, 2011, Proceedings, Part III 15, pp. 596–605. Springer (2011)
9. Hao, X., Shihong, M., Chunyong, Z., Dongyuan, S.: Optimal placement of charging infrastructures for large-scale integration of pure electric vehicles into grid. Int. J. Electr. Pow. Energy Syst. (IJEPES) **53**, 159–65 (2013)
10. He, F., Yin, Y., Zhou, J.: Deploying public charging stations for electric vehicles on urban road networks. Transp. Res. Part C Emerg. Technol. **60**, 227–240 (2015)
11. He, S.Y., Kuo, Y.H., Wu, D.: Incorporating institutional and spatial factors in the selection of the optimal locations of public electric vehicle charging facilities: a case study of Beijing, China. Transp. Res. Part C Emerg. Technol. **67**, 131–148 (2016)
12. Hodgson, M.J.: A flow-capturing location-allocation model. Geograph. Anal. **22**(3), 270–279 (1990)
13. Janjić, A., Velimirović, L., Velimirović, J., Vranić, P.: Estimating the optimal number and locations of electric vehicle charging stations: the application of multi-criteria p-median methodology. Transp. Plan. Technol. **44**(8), 827–842 (2021)
14. Mehar, S., Senouci, S.M.: An optimization location scheme for electric charging stations. In: 2013 International Conference on Smart Communications in Network Technologies (SACONET), vol. 1, pp. 1–5. IEEE (2013)
15. Metais, M.O., Jouini, O., Perez, Y., Berrada, J., Suomalainen, E.: Too much or not enough? planning electric vehicle charging infrastructure: a review of modeling options. Renew. Sustain. Energy Rev. **153**, 111719 (2022)
16. Motoaki, Y.: Location-allocation of electric vehicle fast chargers–research and practice. World Electr. Veh. J. **10**(1), 12 (2019)
17. Owen, S.H., Daskin, M.S.: Strategic facility location: a review. Eur. J. Oper. Res. **111**(3), 423–447 (1998)
18. Rajabi-Ghahnavieh, A., Sadeghi-Barzani, P.: Optimal zonal fast-charging station placement considering urban traffic circulation. IEEE Trans. Veh. Technol. **66**(1), 45–56 (2016)
19. Sun, Z., Gao, W., Li, B., Wang, L.: Locating charging stations for electric vehicles. Transp. Policy **98**, 48–54 (2020)
20. Tian, D., et al.: Optimal location for electric vehicles charging stations based on p-median model. In: Artificial Intelligence and Security: 6th International Conference, ICAIS 2020, Hohhot, China, July 17–20, 2020, Proceedings, Part II 6, pp. 278–289. Springer (2020)

21. Wang, Y.W., Lin, C.C.: Locating multiple types of recharging stations for battery-powered electric vehicle transport. Transp. Res. Part E Logistics Transp. Rev. **58**, 76–87 (2013)
22. Wang, Y.W., Wang, C.R.: Locating passenger vehicle refueling stations. Transp. Res. Part E Logistics Transp. Rev. **46**(5), 791–801 (2010)
23. Yang, J., Dong, J., Hu, L.: A data-driven optimization-based approach for siting and sizing of electric taxi charging stations. Transp. Res. Part C Emerg. Technol. **77**, 462–477 (2017)
24. You, P.S., Hsieh, Y.C.: A hybrid heuristic approach to the problem of the location of vehicle charging stations. Comput. Ind. Eng. **70**, 195–204 (2014)

Road Network and its Impact on Urban Socio-ecological Systems: A Network Perspective

N. V. Neelima[1](✉), Shuanglei Wu[2,3], Shaurya Shriyam[1], and Yongping Wei[2]

[1] Indian Institute of Technology, Delhi 110016, India
qiz238693@iitd.ac.in
[2] The University of Queensland, Brisbane, QLD 4072, Australia
[3] Guangxi University, Nanning 530004, Guangxi, China

Abstract. Road networks are the most basic yet significant permanent infrastructure. With the growing population and demands for urbanization, road networks are increasing intertwined with the natural environment and urban cities. Hence there is a need to understand the dynamics of the road networks and their impacts from a socio-ecological development perspective. This article is a thorough review of 35 articles with the aim to understand and review the various elements that may be impacted by the road infrastructure and the methods adopted by researchers in their study of those elements. It is found that there has been extensive work conducted in road networks from a traffic prediction and management perspective, but understanding of the road network impacts were fragmented and highly constrained to case-by-case studies. Understanding from these highlights the need to leverage network modelling methodologies from a socio-ecological perspective for developing more urban sustainable road systems.

Keywords: Road network modeling · ecological networks · socio-economic networks

1 Introduction

Globally there has been an exceptional rise in the people living in urban areas. It is estimated that by 2050 close to 68% of the world's population will move to an urban settlement [1]. This increase in urbanization requires transportation infrastructure planning and optimization with the consideration of the environmental sustainability. The Sustainable Development Goals (SDGs) proposed by UNDP address transportation issues: SDG 3.6 aims to halve road crash deaths, SDG 11.6 seeks to minimize urbanization's environmental impact, and SDG 13.2 calls for integrating climate change measures into national policies. Road networks are known to be one of the basic infrastructures contributing to a nation's socio-economic development [2]. There has been a co-evolution between road networks and urbanization, for instance, road networks in India have increased by 60% and there has been over 19.6% rise in urbanization in the past decade [3]. The expansion of road networks can directly result in a linear and accretive extension of built-up areas along the main roads [4].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 425–434, 2025.
https://doi.org/10.1007/978-3-031-82439-5_35

There is extensive literature on the impact of road networks on urban socio-economic development since the large-scale development of transportation. These studies include understanding factors for urban expansion [5, 6] and improving traffic specific related issues. The scientific understanding of the ecological effects of roadways is a relatively recent topic. These studies mostly focus on short-term research with narrowly defined goals. To our knowledge, there are limited understanding of the interactive relationships between urban socio-economic development and ecological integrity with the development of road networks. This knowledge gap will compromise our capacity to develop sustainable cities.

Network science is the study of the abstract networking properties of systems and has demonstrating wider applications in research including supply chain economical networks, medical science applications, ecological and road networks. Numerous modelling approaches such as spatial analysis methods and graph theory have been proven effective in studying network evolution based on historical data [10, 11].

This article explores the impact of road networks on urban socio-ecological systems through a network perspective, reviewing literature on modeling and effects, and proposing the need to develop a framework to apply network theory for sustainable urban design and management.

2 Current Research on Road Networks

2.1 Major Themes of Research and Indicators for Network Structures of Roads

The study of road networks integrates various disciplines, including operating systems, economics, mathematics, physics, and transportation processes. Developing transportation infrastructure requires measuring transport activities, linking them to land use dynamics. Scholars use accessibility theory and traffic supply-demand equilibrium theory to explore how infrastructure affects travel choices and urban traffic demand, influencing ongoing system advancements. [12, 13].

Network science has been widely applied to model roads as networks. Strano et al. 2011 [5] and Barthelemy (2011) [14] highlighted that the two processes that take place while expansion of the road networks are densification and exploration. Densification focuses on the existing networks and increasing current capacity whereas exploration involves forming new networks, the former keeps the basic structure intact, and the latter is more responsible for spatial evolution in a region. Network understanding also aids in designing road networks for natural disasters, assessing vulnerability and resilience by evaluating their capacity to absorb disruptions and risks effectively. Table 1 summarizes the network indicators used for road networks.

2.2 Modeling Approaches for Road Networks

Modeling of road networks began in the 1930s, categorizing traffic flow models into three scales: microscopic, macroscopic, and mesoscopic. Microscopic models focus on individual driver behavior, while mesoscopic models bridge the gap using aggregate concepts. Macroscopic models depict traffic as a continuous flow, essential for comprehensive analysis.

Table 1. General indicators of Network analysis and their use in road networks

Literatures	Indicators	Significance for road networks
[15]	Average node degree	Node can be intersection of roads, specific study area
[16]	The topological assortativity coefficient	Provides information on the similarity of nodes in terms of connectivity
[16]	Clustering coefficient	Importance of junctions in roads
[5, 16, 17]	Measures of centrality (Betweenness, closeness and straightness)	Shortest path identification, connectivity
[18]	Network reliability	Used to optimize traffic networks
[19]	Critical node/link	Vulnerability of a road network

Initial modeling approaches involved traffic forecasting relied on probabilistic approaches which are popularly known as parametric models. The models that are included in this category are regression models, Kalman, ARIMA and logistic regression. Figure 1 comprises of the techniques and area of application and salient features for different modeling.

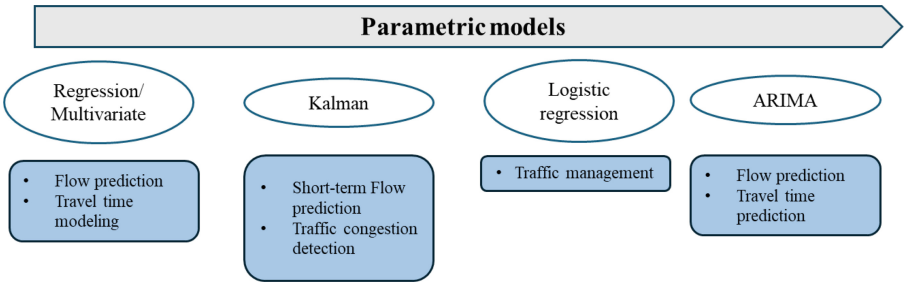


Fig. 1. Parametric models for traffic prediction

The Autoregressive Integrated Moving Average (ARIMA) model is widely used for short-term traffic predictions, especially with large datasets. Parametric models suit smaller datasets, while new technologies in network science address the stochastic nature of traffic flow, necessitating advanced algorithms for complex road networks [8]. Non-parametric techniques thus gained popularity, which involves machine learning algorithms to resolve traffic management issues, travel modeling and congestion. Non-parametric models, primarily, do not assume a specific data distribution, allowing flexibility in applications like traffic management and communication systems. Recent advances include hybrid models like agent-based neural networks, which incorporate qualitative information. In Brisbane, studies found that real-time delay information effectively influenced commuters route choices, with the Neugent hybrid model

improving network performance, although external factors like weather and traffic incidents remain underexplored [6]. Figure 2 shows schematic depiction of commonly used non-parametric models and their applications.

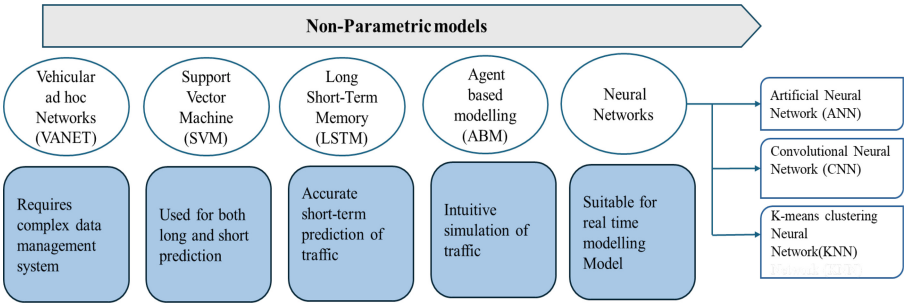


Fig. 2. *N*on-parametric models for traffic prediction

3 Socio-economic Impacts of Road Networks

3.1 Major Themes of Research and Indicators for the Socio-economic Impacts

Transportation networks enable the growth of both social and economic aspects, this happens by the movement of people and goods, significantly driving economic growth. Connectivity enhances socio-economic development, necessitating a deeper understanding of how road network expansion influences overall development [7, 20].

Meanwhile, there are also studies that indicate the socio-economic costs associated with road networks like social cost, which is referred to monetizing any disturbance that is caused in the present scenario due to road network infrastructure [8]. Highway construction disrupts local communities through noise, dust, traffic disruption, and accidents, while social costs during operation include delayed travel, congestion pricing, emissions, and environmental impacts (Marginova et al., 2019) [21]. Mizutani et al. 2011 studied the social cost on Japanese highways, it was found the social cost increases as the traffic volumes increases and traffic congestion and accidents are the major contributors to the social cost [22].

Socio-economic indicators commonly identified are per capita income, GDP, population. Some special indicators have also been used for characterizing road network accessibility which are Australian socio-economic indicators, including Urban and Rural Indices of Relative Socio-Economic Advantage and Disadvantage, Economic Resource Index, and Education and Occupation Index [23].

3.2 Modelling Approaches for the Road Network and its Socio-economic Impacts

Li et al. (2016) emphasized that road networks enhance GDP, draw investments, generate employment, improve rural development, promote urbanization, and alleviate poverty, highlighting their vital importance in China's Western Development Strategy [24]. Hence to understand this causal relationship, some of the models are discussed in this section. The GIS-based model analyzed urban road networks for connectivity and traffic flow, while the cellular automata model simulated local interactions revealing emergent behaviors, and the system dynamics model illustrated how boundary conditions affect network efficiency and long-term urban planning outcomes [15]. The models are listed in Table 2 below.

Table 2. Modelling techniques for socio-economic subsystem

Study	Model	Features
[25]	GIS based	Bottom-up approach, Captures spatial- temporal data, Static in nature
[7, 25]	Cellular Automata	Bottom-up approach, focused on monitoring spatial-temporal changes, Considers an urban area as self-organizing complex system
[25]	System dynamics model	Top-down approach considers urban system as collection of interconnected elements

4 Current Research on Urban Ecology: Ecological Impacts

4.1 Major Themes of Research and Indicators for the Environmental Impacts of Roads

Studies on the interactive impacts between ecological processes and road networks have indicated both positive and negative impacts on the ecosystem and the environment. Amphibian collision studies by Sousa-Guedes et al. (2021) found that geographically weighted regression effectively models animal-vehicle collisions, highlighting species-specific responses to roadways crucial for conservation efforts. [28]. Road networks adversely affect ecosystems by increasing road-stream crossings, disrupting fish migration, and causing habitat degradation [12]. Road ecology researchers advocate for manipulative experiments to strengthen findings, while assessing afforestation feasibility and environmental transport network planning [29]. Green networks are urban landscapes that connect green spaces and water bodies, serving ecological and social functions. They provide habitats for wildlife and promote human movement and recreation. [26]. A study in southwestern Alberta found that grizzly bears adapted to traffic by using low-vehicle-count areas near roads at night, avoiding high-traffic routes, emphasizing the need for better wildlife management practices for accessibility [27].

Quantification of the fragmented habitats by road networks often falls within the scope of landscape ecology. There have been numerous metrics in the ecological network identified by researchers over the years for quantifying ecological impact, some of those have been described in Table 3.

Table 3. Ecological network indices

Studies	Index	Relevance
[30]	Probability of connectivity index	Shows connectivity between two land mass
[31]	Splitting index (SPLIT)	Indicates the tendency of disconnection between patches
[31]	Perimeter-area fractal dimension (PAFRAC)	Shape complexity measurement
[31]	Patch density (PD)	Number of patches of a certain type per landscape area
[32]	Remote Sensing based Ecological Index (RSEI)	To measure the eco-environmental quality of a given area
	Normalized Vegetation Density Index (NVDI)	Indicates state of the ecosystem
	Land Surface Temperature (LST)	Indicates local climatic (i.e., temperature and humidity)
	Land Surface Moisture (LSM)	
[33]	Spatial Road Disturbance Index (SPROADI)	Indicates fragmentation caused by the roads in the ecology
[34]	Integral index of connectivity (IIC)	Integrates various landscape metrics to provide a comprehensive measure of connectivity
	Important value of a patch	Relative change in connectivity index
	Landscape change rate (LST)	Transfer rate between all landscape types in a study period in a region

4.2 Modelling Approaches for the Road Network and its Ecological Impacts

To model the environmental impacts of road networks, including suppression of vegetation, fragmentation of forests, and increased risks of wildlife-vehicle collisions, Table 4. Ecological networks optimization techniques describes different ecological network models and their underlying principles.

Table 4. Ecological networks optimization techniques

Studies	Model	Features
[12, 31]	Minimum Cumulative resistance model	Minimum resistance between patches identified and non-resistance flow of energy and species is prioritized
[31]	Gravity model	Interaction forces between ecological sources is identified and Ecological corridor is suggested basis on this criterion
[4]	Fuzzy comprehensive model	Utilizes fuzzy set theory, incorporating both quantitative and qualitative indicators for evaluation
[4]	Obstacle degree model	Identifies the primary factors that act as obstacles to quantify the gap between the current state and an ideal state
[35]	Corridor cutting model	Identifies risk by analyzing provision of corridor in an ecological space. High ecological risk in the traffic intensive areas

5 Current Challenges and Future Prospects

As reviewed in Sects. 2, 3 and 4 above, the concept of network and network modelling principles apply to the road system and its associated socio-economic and ecological systems. The different indicators used to represent the structure of these networks often have an origin from the commonly used network metrics. This opens an opportunity to develop a network-metrics based model for developing the causal understanding between the development of road networks and its impacts on the associated socio-ecological systems, which is the current knowledge gap in urban transportation studies. However, current understanding is fragmented, and the choice of network indicators and modelling approaches vary in a case-by-case basis. We propose that a framework should be developed to integrate network-based metrics to link road network development with socio-economic growth and ecological integrity. Its global and longitudinal application could reveal spatial patterns and co-evolution of road systems, aiding sustainable city development and supporting related Sustainable Development Goals through data-driven models.

Reaching sustainable development objectives requires more than making efficient use of the resources at hand; it also entails building an environment or system that could support socio-economic development while maintaining the ecological footprint of the area [29]. A conceptual framework should be formed to provide systemic understanding of the interrelated relationships among roads, socio-economic, and eco-environmental development at various temporal and spatial scales. This will not only facilitate network modelling algorithms development in a practical context, but also contribute to formulating solutions for more effective and sustainable road infrastructural planning and management.

Acknowledgement. We gratefully acknowledge the DST Center of Excellence (CoE) in Climate Information, Indian Institute of Technology, Delhi, for the invaluable support and contribution to the accomplishment of this work.

References

1. % of the world population projected to live in urban areas by 2050, says UN | UN DESA |United Nations Department of Economic and Social Affairs. (n.d.). https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html
2. Miao, Z., Pan, L., Wang, Q., Chen, P., Yan, C., Liu, L.: Research on urban ecological network under the threat of road networks—a case study of Wuhan. ISPRS Int. J. Geo Inf. **8**(8), 342 (2019). https://doi.org/10.3390/ijgi8080342
3. Year End Review 2023-Ministry of Road Transport and Highways. https://pib.gov.in/PressReleaseIframePage.aspx?PRID=1993425#:~:text=Road%20Network%20in%20the%20Country,State%20Highways%3A%201%2C79%2C535%20km
4. Wu, X., Hu, F.: Analysis of ecological carrying capacity using a fuzzy comprehensive evaluation method. Ecol. Ind. **113**, 106243 (2020). https://doi.org/10.1016/j.ecolind.2020.106243
5. Strano, E., Nicosia, V., Latora, V., Porta, S., Barthélemy, M.: Elementary processes governing the evolution of road networks. Sci. Rep. **2**(1) (2012). https://doi.org/10.1038/srep00296
6. Dia, H., Panwai, S.: Modelling drivers' compliance and route choice behaviour in response to travel information. Nonlinear Dyn. **49**(4), 493–509 (2007). https://doi.org/10.1007/s11071-006-9111-3
7. Han, J., Hayashi, Y., Cao, X., Imura, H.: Application of an integrated system dynamics and cellular automata model for urban growth assessment: a case study of Shanghai, China. Landscape Urban Plann. **91**(3), 133–141 (2009). https://doi.org/10.1016/j.landurbplan.2008.12.002
8. Tini, N.H., Shah, M.Z., Sultan, Z.: Impact of road transportation network on socio - economic well-being : an overview of global perspective. Int. J. Sci. Res. Sci. Eng. Technol. **4**(9), 282–296 (2018). https://doi.org/10.32628/ijsrset184962
9. Li, C., Gao, X., He, B.-J., Wu, J., Wu, K.: Coupling coordination relationships between urban-industrial land use efficiency and accessibility of highway networks: evidence from Beijing-Tianjin-Hebei urban agglomeration, China. Sustainability **11**, 1446 (**2019**). https://doi.org/10.3390/su11051446
10. Liu, S.L., Cui, B.S., Dong, S.K., Yang, Z.F., Yang, M., Holt, K.: Evaluating the influence of road networks on landscape and regional ecological risk—a case study in Lancang River Valley of Southwest China. Ecol. Eng. **34**(2), 91–99 (2008). ISSN 0925–8574. https://doi.org/10.1016/j.ecoleng.2008.07.006
11. Auttha, W., Klungboonkrong, P.: Evaluation of the transport environmental effects of an urban road network in a Medium-Sized city in a developing country. Sustainability **15**, 16743 (2023). https://doi.org/10.3390/su152416743
12. Mo, W., Wang, Y., Zhang, Y., Zhuang, D.: Impacts of road network expansion on landscape ecological risk in a megacity, China: a case study of Beijing. Sci. Total Environ. **574**, 1000–1011 (2017). ISSN 0048–9697, https://doi.org/10.1016/j.scitotenv.2016.09.048
13. Mundia, C.N., Aniya, M.: Analysis of land use/cover changes and urban expansion of Nairobi city using remote sensing and GIS. Int. J. Remote Sens. **26**(13), 2831–2849 (2005). https://doi.org/10.1080/01431160500117865
14. Barthelemy, M.: Spatial networks. Phys. Rep. **499**(1–3), 1–101 (2011). https://doi.org/10.1016/j.physrep.2010.11.002

15. Cheng, Z., Ouyang, M., Du, C., Zhang, H., Wang, N., Hong, L.: Boundary effects on topological characteristics of urban road networks. Chaos **33**(7) (2023). https://doi.org/10.1063/5.0145079
16. Mukherjee, S.: Statistical analysis of the road network of India. Pramana **79**(3), 483–491 (2012). https://doi.org/10.1007/s12043-012-0336-z
17. Ahmadzai, F., Rao, K.M.L., Ulfat, S.: Assessment and modelling of urban road networks using Integrated Graph of Natural Road Network (a GIS-based approach). J. Urban Manage. **8**(1), 109–125 (2019). https://doi.org/10.1016/j.jum.2018.11.001
18. How to Measure, Test & Improve Network Reliability - Obkio, Obkio. https://obkio.com/blog/how-to-measure-improve-network-reliability/
19. Node.js — Wednesday February 14, 2024, Security Releases. https://nodejs.org/en/blog/vulnerability/february-2024-security-releases
20. Ng, C.P., Law, T.H., Jakarni, F.M., Kulanthayan, S.: Road infrastructure development and economic growth. IOP Conf. Ser. Mater. Sci. Eng. **512**, 012045 (2019). https://doi.org/10.1088/1757-899X/512/1/012045
21. Margorínová, M., Trojanová, M.: Social costs of the road project in the operation phase. Transp. Res. Procedia **40**, 1103–1110 (2019). ISSN 2352–1465, https://doi.org/10.1016/j.trpro.2019.07.154
22. Mizutani, F., Suzuki, Y., Sakai, H.: Estimation of social costs of transport in Japan. Urban Stud. **48**(16), 3537–3559 (2011). https://doi.org/10.1177/0042098011399597
23. Somenahalli, S.V.C., Taylor, M.A.P., Susilawati, S.: Road network accessibility and socio-economic disadvantage across adelaide metropolitan area. Transp. in Dev. Econ. **2**, 15 (2016). https://doi.org/10.1007/s40890-016-0020-y
24. Shiu, A., Li, R., Woo, C.-K.: Economic growth and infrastructure investments in energy and transportation: a causality interpretation of china's western development strategy. Energy J. **37**(1_suppl), 211–222 (2016). https://doi.org/10.5547/01956574.37.si1.ashi
25. Han, J., Yoshitsugu, H., Cao, X., Imura, H.: Application of an integrated system dynamics and cellular automata model for urban growth assessment: a case study of Shanghai. China. Landscape and Urban Plann. **91**, 133–141 (2009). https://doi.org/10.1016/j.landurbplan.2008.12.002
26. Xiu, N., et al.: A socio-ecological perspective of urban green networks: the Stockholm case. Urban Ecosystems **20**(4), 729–742 (2017). https://doi.org/10.1007/s11252-017-0648-3
27. Mace, R.D., Waller, J.S., Manley, T.L., Lyon, L.J., Zuuring, H.: Relationships among grizzly bears, roads and habitat in the swan mountains Montana. J. Appl. Ecol. **33**(6), 1395 (1996). https://doi.org/10.2307/2404779
28. Sousa-Guedes, D., Franch, M., Sillero, N.: A spatial approach for modeling amphibian road-kills: comparison of regression techniques. ISPRS Int. J. Geo Inf. **10**(5), 343 (2021). https://doi.org/10.3390/ijgi10050343
29. Nogués, S., Cabarga-Varona, A.: Modelling land use changes for landscape connectivity: the role of plantation forestry and highways. J. Nat. Conserv. **22**(6), 504–515 (2014). https://doi.org/10.1016/j.jnc.2014.08.004
30. Saura, S., Pascual-Hortal, L.: A new habitat availability index to integrate connectivity in landscape conservation planning: comparison with existing indices and application to a case study. Landsc. Urban Plan. **83**(2–3), 91–103 (2007). https://doi.org/10.1016/j.landurbplan.2007.03.005
31. Huang, X., Wang, H., Shan, L., Xiao, F.: Constructing and optimizing urban ecological network in the context of rapid urbanization for improving landscape connectivity. Ecol. Ind. **132**, 108319 (2021). https://doi.org/10.1016/j.ecolind.2021.108319
32. Xu, H.Q., Wang, M.Y., Shi, T.T., Guan, H.D., Fang, C.Y., Lin, Z.L.: Prediction of ecological effects of potential population and impervious surface increases using a remote sensing based ecological index (RSEI). Ecol. Indic. **93**, 730–740 (2018)

33. Nematollahi, S., Fakheran, S., Kienast, F., et al.: Application of InVEST habitat quality module in spatially vulnerability assessment of natural habitats (case study: Chaharmahal and Bakhtiari province, Iran). Environ. Monit. Assess. **192**, 487 (2020). https://doi.org/10.1007/s10661-020-08460-6
34. Cui, L., Wang, J., Sun, L., Lv, C.: Construction and optimization of green space ecological networks in urban fringe areas: a case study with the urban fringe area of Tongzhou district in Beijing. J. Clean. Prod. **276**, 124266 (2020). https://doi.org/10.1016/j.jclepro.2020.124266
35. Yan, Z., You, N., Wang, L., Lan, C.: Assessing the impact of road network on urban landscape ecological risk based on corridor cutting degree model in Fuzhou, China. Sustainability **15**(2), 1724 (2023). https://doi.org/10.3390/su15021724
36. Havlin, S., et al.: Challenges in network science: applications to infrastructures, climate, social systems and economics. Eur. Phys. J. Spec. Top. **214**(1), 273–293 (2012). https://doi.org/10.1140/epjst/e2012-01695-x
37. Magness, D.R., Morton, J.M., Huettmann, F., Chapin, F.S., McGuire, A.D.: A climate-change adaptation framework to reduce continental-scale vulnerability across conservation reserves. Ecosphere **2**(10), art112 (2011). https://doi.org/10.1890/es11-00200.1
38. El Gouj, H., Rincón-Acosta, C., Lagesse, C.: Urban morphogenesis analysis based on geo-historical road data. Appl. Netw. Sci. **7**(1) (2022). https://doi.org/10.1007/s41109-021-00440-0
39. T. R. B. and N. R. Council, assessing and managing the ecological impacts of paved roads (2005). https://doi.org/10.17226/11535

A Complex Networks Approach to Evaluate the 15-Minute City Paradigm and Urban Segregation at Different Scales

Mirko Lai[1], Salvatore Vilella[1](✉), Federica Cena[2], and Giancarlo Ruffo[1]

[1] Dipartimento di Scienze e Innovazione Tecnologica, Università del Piemonte Orientale, Alessandria, Italy
salvatore.vilella@gmail.com

[2] Dipartimento di Informatica, Università degli Studi di Torino, Turin, Italy

Abstract. This paper proposes to exploit complex network analysis to evaluate service accessibility within urban environments in accordance with the 15-minute city paradigm. Our approach introduces an analytical pipeline that includes the quantification of urban segregation and allows for detailed comparisons between different cities and distinct urban areas.

The 15-minute city concept emphasizes the importance of having fundamental services and amenities accessible within a 15-minute walk or bike ride, aiming to enhance urban livability and sustainability. By modeling cities as complex networks, we analyze various metrics to capture the intricate relationships and accessibility patterns within urban spaces. Our methodology assesses the connectivity and distribution of PoIs, the transportation networks, and the residential areas, providing insights into urban density and diversity. Furthermore, we suggest to use Closeness centrality as a metric to quantify urban segregation, also highlighting how disparities in access to essential services can be easily interpretable by means of other socioeconomic inequalities. The methodology is showcased on two case studies, i.e., the Italian cities of Milan and Turin.

Keywords: connectivity · 15-minute city · urban mobility · community detection · segregation

1 Introduction

The 15-minute city concept [7] is a widely adopted strategy that some cities are following to enhance the livability and inclusiveness of urban areas. This concept emphasizes designing neighborhoods where residents can reach all essential services and amenities within a 15-minute walk or bike ride from their homes. The goal is to minimize reliance on cars, encourage healthier lifestyles, and promote sustainability. However, some studies raise concerns that ensuring inclusive access to all necessary services within a neighborhood might inadvertently lead to gentrification, potentially displacing lower-income residents and fostering segregation [1].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 435–446, 2025.
https://doi.org/10.1007/978-3-031-82439-5_36

Our goal is to introduce an analytical pipeline that enables city planners, stakeholders, and ordinary citizens to investigate how the fitness to the 15-minute city concept is linked to urban segregation, inequality and ghettoization. We claim that complex networks offer the optimal framework for establishing this pipeline. Representing the urban landscape by means of network's metrics is a widely adopted approach based on building a directed graph where nodes are streets intersections and edges are roads' segments connecting nodes (e.g., [3,9]). Using such general framework, accessibility to Points of Interest (PoIs) can be measured in terms of time traveled to reach them from a given starting point, following an existing path in the graph. Links should be walkable or bikable (e.g., as in [8]), if we want to calculate our metrics in the pedestrian (or the cycling) network; conversely, to analyze the urban transport network (e.g., as in [4]), we need the edges in the graph to be served by buses, metro, local trains, etc.

In our approach, we aim to assess how well amenities are accessible in cities in the pedestrian network, and if hyper-proximity is correlated with urban segregation additionally considering the public transit network. To do so, we reduce many well-known measures to network-based quantitative dimensions. Taking inspiration from the 15-minute city paradigm, we reformulated the classic notions of *proximity*, *density* and *diversity*, that can be defined at different scales: the city as a whole, the neighborhoods, and the individual residential addresses. For an exhaustive analysis of the many neighborhood metrics that have been used in the literature to study city walkability and hyper-proximity, see [2].

We have access to other publicly available socio-demographic data that can provide insights of the heterogeneous distribution of such metrics in cities, giving stakeholders and policy makers guidance on how urban planning can help to reduce social and physical distances in cities and improve citizens' quality of life.

Our Contribution: The most relevant papers are connected to computational platforms that calculate accessibility metrics to compare cities and neighborhoods with each other, such as the CityChrone Project [4] and the City Access Map [8]. Our approach is complimentary to both, since we use complex networks analysis for computing walking accessibility to services and also connectivity to the rest of the city by urban transport.

Our approach is showcased with two cities, Milan and Turin, which are shown in [8] as outperforming other major cities in the world under the 15-minute concept. We can analyze cities at different scales proving that there is a very strong correlation between accessibility to services and connectivity. Using our methodology, it is easy to identify urban areas that are mostly disadvantaged on both dimensions, or where this correlation does not exist. A deeper analysis of these particular neighborhoods shows that reasons for segregation or lack of services may be very different from each other.

2 Data

Our primary source is *OpenStreetMap* (OSM), a free world crowd-sourced map, we used to retrieve the pedestrian road network and the positions of PoIs for the two case studies. The roads and PoIs network allowed us to compute the physical distances among residents and nearest services and amenities, as well as from other neighborhoods. It should be noted that it is well known that OSM does not offer homogeneous coverage across all areas. However, it is still an important open data sources that allows us to set up comparative analyses on many different cities.

Additionally, we employed census tract data from the *2021 Italian Census* provided by ISTAT[1] to map the socio-economic landscape of the two cities. This dataset offers a comprehensive snapshot of the population, including key indicators such as age distribution, gender, and nationality. By integrating this demographic information with our spatial analysis, we were able to explore how urban structure intersects with population characteristics, revealing potential disparities in access to essential services.

To complement these datasets, we used public transportation information available in *GTFS* format from the local open data portals of Turin and Milan[2].

3 Methods

3.1 PoI's Proximity, Density, Entropy as Node-Wise Measures

To assess accessibility to services from the perspective of a single node, we first need to assign a category to each PoI. Although there is no agreement among scholars on which and how many categories to use to make such aggregation, neither which services to include in each category, here we adopt an updated version of the classification proposed in [8], that uses seven categories: *Mobility*, *Active Living*, *Entertainment*, *Food Choices*, *Community Space*, *Education* and *Health and Well-being*. In the following, we refer to this set of categories as CAT; also, given a city c, and the set of overall PoIs in c $\mathrm{POI}_c = \{p_1, p_2, \ldots\}$, we map every PoI $p \in \mathrm{POI}_c$ to one of the above mentioned categories with notation $\mathrm{cat}(p)$.

For each city c, we build a *pedestrian network*, that is a weighted directed graph $G_c^{\mathrm{ped}} = (N_c, E_c)$, where each node $n \in N_c$ is an intersection, and an edge $e_{ij} \in E_c : e_{ij} = (n_i, n_j)$ is a walkable street segment between two nodes n_i and n_j. Every PoI is assigned to the closest node in G_c^{ped}: this can be formalized with function $\mathrm{poi}(p) = n_p$ that maps each $p \in \mathrm{POI}_c$ to a node $n_p \in N_c$. Similarly we can assign every residential address to the closest node in the pedestrian network. Moreover, every edge is weighted by the distance in meters from one endpoint to the other $d(e_{ij})$, and the average time $t(e_{ij})$ to walk from n_i to n_j.

[1] https://shorturl.at/8fFV3.
[2] Turin: https://shorturl.at/9Eush, Milan: https://shorturl.at/Jot8y.

Given two nodes $n, m \in N_c$, they are connected if exists a path in G_c^{ped}. Hence, we can define $d(n, m)$ as the *spatial distance* between n and m, i.e., the length of the minimum shortest path in meters, and $t(n, m)$ as the *time distance* between n and m, i.e., the length of the minimum shortest path in minutes. In general, we can refer to $d(x, y)$ or $t(x, y)$ as the spatial or time distance between x and y, whatever they are intersections, PoIs, or residential addresses.

Analogously, we can build a *cycling network* for each city, but for the sake of simplicity in this paper we will make use of the pedestrian network only to estimate accessibility to services. This representation allows us to calculate the walkable distance, both in meters and in time travel (assuming an average velocity of 5 km/h), from any point of the city to the other. The reader might also notice that we can add precision to our model storing for each PoI $p \in \text{POI}_c$ and for every residential address the actual distances in meters and in walking time to the node n they were assigned to. However, in this paper we will simplify our analysis without considering this additional distances.

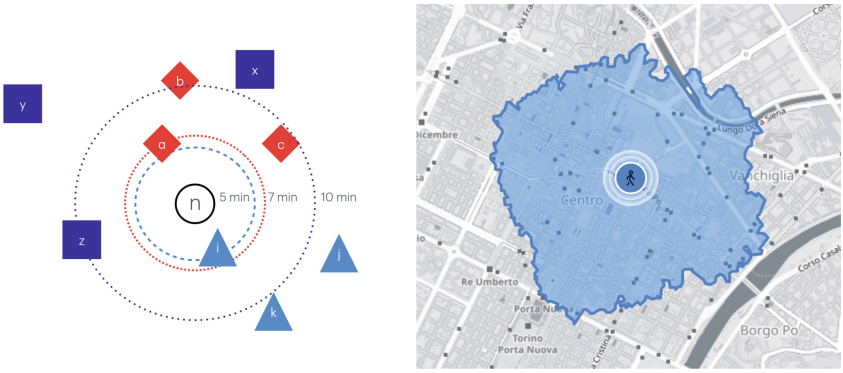


Fig. 1. On the left, we have node n and its closest PoIs, three for each of some given abstract categories: azure triangles, red diamonds, and blue squares. For each of these categories, the closest PoIs are i (at walkable distance of 5 min), a (7 min), and z (10 min). This means that $\mathcal{P}(n) = 10$, because within - and no less than - 10 min residents in n can walk to a PoI for each of the given categories. On the right, we show the isochrone $\mathcal{I}(15, n)$, where n is one of the intersections of Piazza Castello in Turin, Italy. The map has been created by the TravelTime Api.

We re-interpreted the most common definitions of the 15-minute city concept (see [7]) to quantitative network metrics to get a three-dimensional estimate of accessibility to services.

First, let's recall that the proximity dimension should assess if basic services are readily available to residents within a 15-min radius.

Definition 1 (PoI-proximity). *Given node $n \in N_c$, the PoI-proximity $\mathcal{P}(n)$ is the minimum temporal radius needed by residents in n to reach at least one*

PoI for every category in CAT, i.e., $\mathcal{P}(n)$ *is the minimum temporal distance such that* $\forall x \in CAT, \exists p \in POI_c : (cat(p) = x \wedge \mathcal{P}(n) \leq t(n,p))$.

For example, if $\mathcal{P}(n) = 10$, as in Fig. 1 (left), then residents assigned to n will find at least one PoI for every category within 10 min; also, this temporal distance is minimal, meaning that at least one category is not accessible through shorter walks, e.g., of 9 min.

PoI-Proximity returns for every node in the graph its de-facto 'hyper-proximity' time parameter. For example, if $\mathcal{P}(n) \leq 15$ for every node n in N_c, then we might be tempted to say that c is a 'perfect' 15-minute city. This is not totally right: on one side, if $\mathcal{P}$ is high on average, the city might be far from respecting the hyper-proximity condition. However, $\mathcal{P}(n)$ alone is not a good measure to draw conclusions: for instance, we can have only one PoI for each category at small walkable distance, that may be problematic basically for two reasons: (1) the number of services in the area should be proportional to residents leaving nearby, and (2) every category in CAT can include a quite heterogeneous number of tags, corresponding to very different kind of services, e.g., both 'hospital' and 'pharmacy' are members of the 'Health and Well-being' class according to OSM. Hence, we suggest using PoI-proximity together with other measures that, after an area around a given node n has been identified, estimate how many PoIs are present inside that area (PoI-density), and also how many PoIs of different categories are reachable within a walk of t minutes from n (PoI-entropy).

To define these other two measures, we will make use of an isochrone map to describe a region accessible from a given geographical point. These maps have been used since 1881 [5], and they are made of contours marking the area reachable within a certain time threshold (see an illustrative example in Fig. 1 (right)).

Definition 2 (Isochrone). *Given node* $n \in N_c$*, the isochrone* $\mathcal{I}(t,n)$ *is a geographical area whose internal points are walkable from* n *in at most* t *minutes. If* $t = \mathcal{P}(n)$*, we can simply refer to the isochrone of* n *with* $\mathcal{I}(n)$.

Given an area, under the 15-minute city concept, density returns the number of people per kilometer square. Here, we want to focus on the density of PoIs per isochrones, instead.

Definition 3 (PoI-density). *Given a node* $n \in N_c$*, a time* t *in minutes, the PoI-density* $\mathcal{D}(t,n)$ *is the ratio between the number of PoIs in isochrone* $\mathcal{I}(t,n)$ *and the area in kilometer square of* $\mathcal{I}(t,n)$*. If* $t = \mathcal{P}(n)$*, we can simply refer to the PoI-density of* n *with* $\mathcal{P}(n)$.

It is possible that, for certain regions, the number of PoIs is greater than areas in kilometer square. To make comparisons easier and to keep $\mathcal{D}(t,n)$ in a range of $[0,1]$, we apply a min-max normalization.

Finally, the third most important 15-minute city dimension is density, whose goal is to quantify the mix of residential, commercial and entertainment components (the highest the better) in a given area. We interpret this concept calculating how PoIs in a given isochrone are distributed for each category.

Definition 4 (PoI-entropy). *Given a node $n \in N_c$, a time t in minutes, the PoI-entropy $\mathcal{E}(t,n) = -\sum_{x \in CAT}(Pr(x) \cdot \log Pr(x))$, where $Pr(x)$ is the probability that a PoI p within the isochrone $\mathcal{I}(t,n)$ belongs to category x (i.e., $cat(p) = x$).*

This dimension calculates the entropy of the distribution of categories for points within the isochrone defined by t and n. High entropy might indicate a mixed-use area with a variety of amenities and services, while low entropy might indicate specialized zones (e.g., residential or industrial areas).

It should be noted that diversity usually aims to measure also heterogeneity in culture and people. Information such as diversity in ethnicity, education, and employment can be calculated using socio-demographic official data for each geographical region. In the rest of the paper, we make use of both terms diversity and entropy to indicate respectively social heterogeneity and variety of accessible services.

As a final observation, PoI-proximity, PoI-density, and PoI-entropy are three different node-wise measures of accessibility to services. Other socio-demographic information would help to estimate accessibility to services: e.g., people density and diversity. Other authors (e.g., see [4,8]) prefer to use only one accessibility measure to better analyze correlations between dimensions. On the contrary, we adopted different measures to allow the implementation of a fine-grained analytical tool, and to provide more informative suggestions to policy makers and stakeholders willing to invest on urban planning.

The three measures provide us with information about the accessibility to services and amenities within a city, and that are complementary from a semantic point of view. We can combine these measures into a single linear combination that returns a combined **PoI-Accessibility** score:

$$\mathcal{A}(n) = w_1 \mathcal{P}'(n) + w_2 \mathcal{D}'(t,n) + w_3 \mathcal{E}'(t,n) \tag{1}$$

where w_1, w_2, and w_3 are the weights that can be attributed to the corresponding measure, t is set to a fixed time in minutes, while the apex indicates that the measures have been normalized by *minmax* scaling. For our analysis, we simply set $w_1 = w_2 = w_3 = 1/3$, and $t = 15$.

Let us note that we aim to define a general measure where values approach 0 when accessibility is low and 1 when accessibility is high. To achieve this, we normalized PoI-Proximity differently from PoI-Density and PoI-Entropy. because higher PoI-Proximity indicates that one or more PoIs of certain categories are further from a given intersection, so that accessibility is lower. As a result, the normalized PoI-Proximity $\mathcal{P}'$ is defined as it follows:

$$\mathcal{P}'(n) = \frac{\max_n \mathcal{P}(n) - \mathcal{P}(n)}{\max_n \mathcal{P}(n) - \min_n \mathcal{P}(n)} \tag{2}$$

As we will see, while the combined accessibility score is a useful indication that gives us an overall evaluation of the services reachability of a city, it remains important to focus and analyze the individual metrics of proximity, density and entropy to explore weaknesses in urban planning at a finer granularity.

3.2 Using Closeness Centrality to Measure Distances Within a City

If we want to assess if amenities and services are at a reasonable walkable distance from any point of the city, at the same time we aim to evaluate if the same point is well connected to the rest of the city, allowing citizens leaving there to easily reach other urban spaces and other people.

Hence, to study connectivity of a node w.r.t. the rest of the city c, we build also the *urban transport network*, that is a weighted directed graph $G_c^{\text{urb}} = (N_c, L_c)$, where a link $l_{ij} \in L_c : l_{ij} = (n_i, n_j)$ is a street segment between two nodes n_i and n_j served by the urban transport service (e.g., buses, metro, local trains, etc.). Then, to understand how segregation may be varying in a city, we calculate the closeness centrality scores for each node in G_c^{urb}.

Definition 5 (Closeness). *The (normalized) closeness of node n is calculated as $\mathcal{C}(n) = \frac{|N_c|-1}{\sum_{m \neq n : m \in N_c} t(n,m)}$, where $t(n,m)$ is temporal distance between n and m in the urban transport network G_c^{urb}.*

In this domain, closeness centrality measures how an intersection (and the PoIs and residential addresses mapped to the corresponding node) is close to all the other intersections of c, where $t(n,m)$ is the average temporal distance that it takes to go from a station mapped to n to another station mapped to m by means of the public transport system represented by G_c^{urb}. Higher normalized closeness $\mathcal{C}(n)$ indicates that n is well connected to the rest of the city, lower $\mathcal{C}(n)$ indicates that n is likely to be in a more segregated part of the city. On a side note, we calculated the temporal distance $t(n,m)$ by averaging the travel times across all scheduled trips on the same line, as well as across different lines that serve the same stations).

3.3 Measuring Accessibility and Connectivity for Multi Scale Regions

We can easily extend the node-wise metrics we defined in Sects. 3.1, and 3.2 to urban regions at different scales: given a region (i.e., census, administrative district, network cluster, city as a whole), we can average PoI-proximity, PoI-density, PoI-entropy values, and closeness centrality scores, as well as plotting their distributions, calculating the variance, and so on. We expect that these distributions will show evidence of inequalities in accessibility to services and connectivity between different parts of the city, and between cities.

These values can also be compared with socio-demographic information for census areas, administrative districts, and cities, to find signals of ghettoization, and correlations with our estimates of connectivity (urban segregation), and accessibility to services (for example, under the 15-minute concept).

Another possible aggregation is looking for 'natural' neighborhoods, by running a community detection algorithm on the pedestrian network G_c^{ped}. We used the Infomap community detection algorithm [11] to find a city's neighborhoods.

We chose this method because it works well to reveal clusters in weighted and directed networks, and it uses the probability flow of random walks to decompose the network into modules by compressing a description of the previously calculated probability flow. This approach models the behavior of a walker that, at each intersection, takes a street randomly to reach another intersection. If some nodes are tightly coupled together, they are likely to be traversed more than once by the same walker: when the walking process ends, it returns a modular and natural structure of the underlying street network. As observed in [6], Infomap provides a city-agnostic method for detecting natural neighborhoods, and also to untie the quantitative analysis from probably outdated urban strategies that defined administrative neighborhood boundaries years ago.

These measures can be easily compared with each other looking for correlations and patterns. For example, we can use bubble charts to compare PoI-Accessibility and Closeness: we can easily spot census areas (Fig. 3, right column) or entire neighborhoods (Fig. 4), that are more disconnected from other parts of the cities (left part of the diagrams), or showing lower accessibility to services (bottom part of the diagrams). Bubble sizes can be drawn proportionally to the population and the color code may be applied to another categorical information, such as the neighborhoods found by Infomap.

4 Analyzing Accessibility and Segregation: The Case Studies of Turin and Milan

Given a set of cities $\{c_1, c_2, c_3, \ldots\}$, we can create graphs $G_{c_i}^{\text{ped}}$ and $G_{c_i}^{\text{urb}}$ for all i. Then, we can calculate some basic statistics over $\mathcal{P}(n)$, $\mathcal{D}(t,n)$, $\mathcal{E}(t,n)$ and $\mathcal{A}(n)$, where t is a fixed hyper-proximity temporal parameter (e.g., 15 min), and n is a node of $G_{c_i}^{\text{ped}}$. Similarly, we can calculate $\mathcal{C}(n)$ for all the nodes n of $G_{c_i}^{\text{urb}}$. If distributions allow to consider the single average as a scale of that measure for the given city, then we are allowed to compare cities by means of these values. As a result, we can rank cities according one or more of these measures. A similar operation has been done in [8], showing that Milan and Turin, among the largest cities in Europe, have the best performances in terms of service accessibility according the 15 min city concept on a worldwide comprehensive analysis.

Focusing on Milan and Turin as the case studies to showcase our analytical methodology, we aim to show that even in 'model' cities, there are important differences between urban areas, and that some interesting links to public transit connectivity may emerge, suggesting how to eradicate or prevent the formation of ghettos. In particular, distributions of $\mathcal{P}(n)$, $\mathcal{D}(15,n)$, and $\mathcal{E}(15,n)$ for both cities are shown in Fig. 2. As discussed in the previous section, we run Infomap on the corresponding pedestrian networks to find the 'natural' neighborhoods, to find 6 network communities in Turin, and 9 in Milan (see the left column of Fig. 3). Then, we aggregate our metrics for every administrative census areas or tracts (4,044 in Turin, and 5,937 in Milan). producing the PoI-Accessibility and Closeness distributions that are shown on the right column of Fig. 3. Also, we can assign each census area to a larger neighborhood, if the majority of

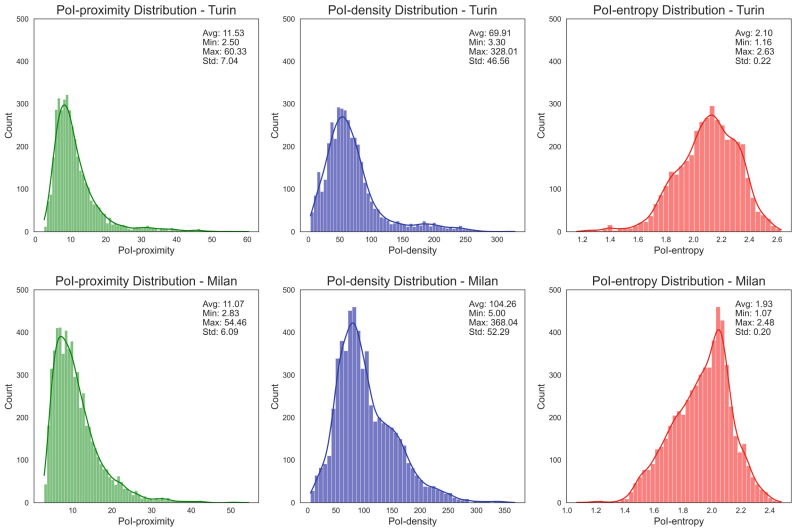
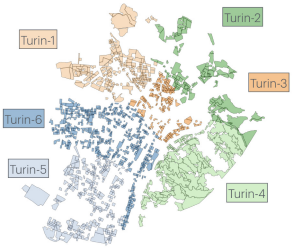


Fig. 2. Distributions of $\mathcal{P}(n)$, $\mathcal{D}(15, n)$, and $\mathcal{E}(15, n)$ for the Italian cities of Turin (top) and Milan (bottom). The solid lines represent empirical distributions smoothing.

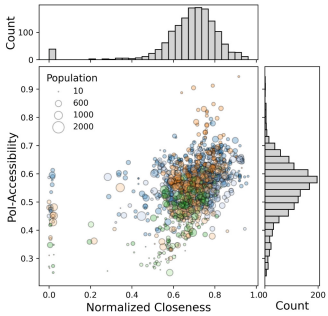
nodes in a census area have been mapped to a that given network community by Infomap. Quite interestingly, neighborhoods found by Infomap in the pedestrian networks fit quite well with the official municipalities of Milan[3] and Turin[4]. The right column of Fig. 3 is also illustrative of the heterogeneous values that PoI-Accessibility and Closeness may get in a city: although the majority of the census areas fall in the top-right part of the bubble charts (i.e., low segregation and high accessibility), showing once more that Turin and Milan are 'model' city w.r.t. the 15-minute concept and overall public transit connectivity, we can also easily spot some areas with very low connectivity (on the left of the bubble charts), and some other areas that perform poorly in terms of accessibility to services within 15 min isochrones (on the bottom of the bubble charts).

These inequalities emerge more clearly if we isolate the neighborhoods found by Infomap, as in Fig. 4. In fact, in Turin we have three neighborhoods (Turin-1, Turin-2, Turin-4) that have a tendency to the bottom-left part of the diagram, meaning that we have a significant number of nodes in those areas of the city that show distinctly both lower connectivity and lower PoI-accessibility than the other three neighborhoods. In Milan we can see that the city center (Milan-5) is very well connected to the rest of the city and it performs very well according the 15-minute accessibility metrics, while another well populated area (Milan-9) has a much lower PoI-Accessibility score. Also, if in Milan we easily spot an interesting, although probably not surprising, core-periphery pattern (i.e., while moving away from the city center, both PoI-Accessibility and Closeness tend to

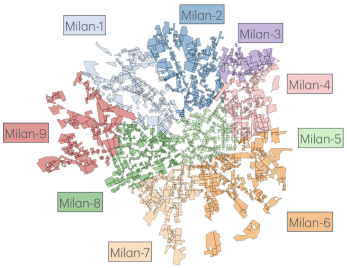
[3] See https://en.wikipedia.org/wiki/Municipalities_of_Milan.
[4] See https://it.wikipedia.org/wiki/Circoscrizioni_di_Torino.



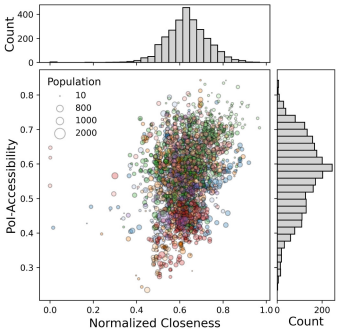
(a) Map of Turin census areas reached by public transport.



(b) $\mathcal{A}$ vs $\mathcal{C}$ - Turin



(c) Map of Milan census areas reached by public transport.

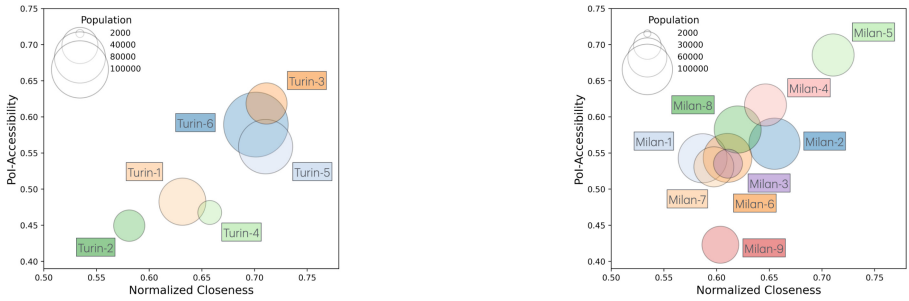


(d) $\mathcal{A}$ vs $\mathcal{C}$ - Milan

Fig. 3. PoI-Accessibility vs Closeness distributions in the cities of Turin (top) and Milan (bottom). Nodes are color-coded according the neighborhoods detected with Infomap.

decrease), in Turin we observe that there are well served peripheries (e.g., Turin-5), as well as other peripheral areas whose urban connectivity and accessibility to services can be definitely improved (e.g., Turin-1, Turin-2, Turin-4).

It is important to stress out that connectivity and accessibility evaluations must be interpreted together with other socio-demographic information to get a better understanding of urban areas that risk ghettoization, that should not be confused with voluntary isolation (as it happens in the so called 'gated communities'). Data about immigration rates could be used as measures of the cultural diversity - or, conversely, homogeneity - of an area, while information about higher education rates are well known to be positively correlated with GDP and with many indicators of quality of life [10]. Let us analyze Turin's neighborhoods to provide an illustrative example (see Table 1). If we look at the immigration and higher education rates that characterize the three neighborhoods of Turin



(a) $\mathcal{A}$ vs $\mathcal{C}$ - Turin (b) $\mathcal{A}$ vs $\mathcal{C}$ - Milan

Fig. 4. PoI-Accessibility vs Closeness by neighborhoods in Turin (left) and Milan (right) aggregated. Nodes are color-coded as in Fig. 2

mentioned above, we can observe opposite situations; in fact, Turin-1 and Turin-2 have conspicuously higher immigration and lower tertiary education rates than Turin-4.

Table 1. Turin's socio-demographic data by neighborhood

Neighborhood	Immigration rate	Tertiary education rate
Turin-1	*14%*	*10%*
Turin-2	*13%*	*10%*
Turin-3	20%	17%
Turin-4	**08%**	**36%**
Turin-5	10%	18%
Turin-6	11%	26%
Turin	13%	19%

This suggests that the inhabitants of disconnected and poorly served areas are not necessarily similar from a socio-demographic point of view. In particular, it is well known that the majority of luxury houses in Turin can be found in the eastern part of the city, beyond the Po river and above the Superga hill: here, residents are likely to use cars more than public transport or just walking, and to keep a privileged disconnection from the rest of the city. Conversely, this is unlikely to be the case for citizens of Turin-1 and Turin-2, who may be struggling to obtain a better service in terms of urban connectivity and proximity to relevant PoIs. Therefore, signals of segregation can be interpreted as tendencies towards either ghettoization or neighborhood exclusivity and voluntary self-isolation, two quite opposite urban categories that should be treated

differently by urban planners and policy makers looking for the right strategies to improve cities' livability and inclusiveness.

5 Conclusions and Future Work

In this study, we propose a network-based methodology to assess urban accessibility to services and public transport connectivity, formalizing the concepts of PoIs proximity, diversity and entropy within a network analysis framework. Our approach has been showcased in two 'model' city (i.e., Milan and Turin): we find that, although there is a clear relationship between accessibility to services at walking distance and connectivity to the rest of the city through the public transit systems, the reasons behind segregation of some urban areas can vary between neighborhoods. We have indications that distinct socio-demographic factors may be causing some areas to be isolated and/or have limited access to services. Future work will involve incorporating this methodology to our United-and-Close platform [6] to analyze a broader set of cities, and also to produce more personalized accessibility and connectivity analyses that are tailored to specific citizens' habits and characteristics.

Funding Information. This research has been partially funded by PON "Ricerca e Innovazione" 2014–2020 framework, and by the European Union - Next Generation EU, Mission 4 Component 2 - CUP C53D23005810006.

References

1. Abbiasov, T., et al.: The 15-minute city quantified using human mobility data. Nat. Hum. Behav. **8**(3), 445–455 (2024)
2. Agampatian, R.: Using GIS to measure walkability: a case study in New York city. Master's thesis, KTH, Geodesy and Geoinformatics (2014)
3. Barthélemy, M.: Spatial networks. Phys. Rep. **499**(1–3), 1–101 (2011)
4. Biazzo, I., Monechi, B., Loreto, V.: General scores for accessibility and inequality measures in urban areas. Roy. Soc. Open Sci. **6**(8), 190, 979 (2019)
5. Galton, F.: On the construction of isochronic passage-charts. Proc. R. Geogr. Soc. Mon. Rec. Geogr. **3**, 657–658 (1881)
6. Lai, M., Vilella, S., Cena, F., Patti, V., Ruffo, G.: United-and-close: an interactive visual platform for assessing urban segregation within the 15-minutes paradigm. In: in Adjunct Proceedings of ACM UMAP '23, pp. 115–120 (2023)
7. Moreno, C., Allam, Z., Chabaud, D., Gall, C., Pratlong, F.: Introducing the "15-minute city": sustainability, resilience and place identity in future post-pandemic cities. Smart Cities **4**(1), 93–111 (2021)
8. Nicoletti, L., Sirenko, M., Verma, T.: Disadvantaged communities have lower access to urban infrastructure. Environ. Plan. B: Urban Anal. City Sci. **50**(3), 831–849 (2023)
9. Porta, S., Crucitti, P., Latora, V.: The network analysis of urban streets: a dual approach. Phys. A **369**(2), 853–866 (2006)
10. Roser, M., Ortiz-Ospina, E.: Human development index (HDI) (2024). https://ourworldindata.org/human-development-index. Accessed 03 Sept 2024
11. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. Proc. Natl. Acad. Sci. **105**(4), 1118–1123 (2008)

Complex Network Analysis as a Tool for Road Maintenance Management

Annamaria Ficara[1(✉)], Giuseppe Sollazzo[2], Gaetano Bosurgi[2], and Maria Fazio[3]

[1] Department of Cognitive Science, Education and Cultural Studies, University of Messina, Via Concezione 6/8, 98121 Messina, Italy
aficara@unime.it

[2] Department of Engineering, University of Messina, Contrada di Dio, 98158 Sant'Agata, Messina, Italy

[3] Department of Mathematics, Computer Science, Physics and Earth Science, University of Messina, Viale F. S. d'Alcontres 31, 98166 Messina, Italy

Abstract. "Smart Roads" represent innovative and integrated systems that evolve the physical road infrastructure by leveraging several sensors and innovative powerful data analysis capabilities, aiming to improve safety and comfort for users. However, agencies lack management systems to leverage smart road data to control infrastructure assets, component conditions, and operational quality. This paper aims to fill this gap by proposing a new approach for road network assessment and characterization based on Complex Network Analysis. This offers a novel perspective to road administrators in decision-making processes, allowing more exhaustive knowledge of the network structure and ensuring support in the evaluation of maintenance strategy ranking.

Keywords: Smart Roads · Road Maintenance · Road Network · Complex Network Analysis

1 Introduction

The ongoing digital transition affecting all the aspects of our everyday lives is even strongly transforming the transport sector by enabling Smart Roads infrastructures and services [17]. By using distributed sensors, modern technologies and advanced analysis capabilities, Smart Roads ensure their mutual interaction with the external context and users for deploying innovative and useful services (for maintenance, operations, safety, etc.) [8]. A critical aspect of Smart Roads is the management of the overall road network, especially considering issue detection and maintenance planning. Because of the increasing data flow from the several heterogeneous sensors deployed along Smart Roads and the computing resources necessary to store and analyze such data, traditional road management approaches appear inadequate, and thus there is a need for defining more innovative and flexible solutions [5]. This paper proposes a new strategy

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 447–458, 2025.
https://doi.org/10.1007/978-3-031-82439-5_37

for providing decision support to policy-makers by leveraging Complex Network Analysis (CNA). In the proposed solution, the road network is modelled as a graph and centrality measures from CNA are applied to find the most central intersections and roads that present critical situations.

There are several approaches in the literature for applying CNA to road networks. Holme [12] simulated simple traffic flow models studying the relation between centrality on a static network, and the way the network structure affects the speed of the traffic flow. Park and Yilmaz [16] introduced a social network analysis approach using entropy and centrality to obtain information on existing road networks to help transportation experts. Jayaweera [13] used four centrality measures (e.g., degree, closeness, betweenness and eigenvector) to identify (i) the most important nodes and the road segments, and (ii) the traffic congestion of a road network in Sri Lanka. Kirkley *et al.* [14] conducted a large-scale empirical study of the betweenness centrality across 97 of the world's largest cities as measured by population. Ahmadzai *et al.* [2] modeled a Kandahar (Afghanistan) road network proposing a method named Integrated Graph of Natural Road Network and measuring closeness, betweenness and straightness. Pung *et al.* [18] presented an efficient disruption framework based on graph theory. This framework was able to simplify road networks preserving their key topological characteristics.

Our approach is innovative because pushes into the CNA solutions, such as the centrality metrics based on the local or global structure of the network, together with insights derived from the data collected by sensors in Smart Roads. Starting from an innovative architecture for road anomaly detection [8], which deploys cloud-based digital services for supporting road network-scale asset management, in this paper we discuss how using CNA to evaluate road network features and to support road administrators in pre-screening the asset condition on which establish and plan more accurate survey activities, for an effective maintenance and rehabilitation optimization. The use of centrality measurements in graphs can also be enriched by time-dependent attributes (i.e., attributes related to road conditions, traffic intensity, weather conditions, etc.) to offer a more powerful tool to road administrators for ensuring more exhaustive knowledge of the network structure and adequate support in the evaluation of maintenance strategy ranking.

2 Reference Scenario

This research work proposes an innovative methodology for the implementation of an analysis tool in a Road Maintenance Management System (RMMS). An RMMS is a digital ecosystem based on a network of sensors deployed along the Smart Roads, which collect data about cars, traffic and the infrastructure. Such data are sent to a central computing system (e.g., a Cloud datacenter) to implement a "first-level analysis", which aims to identify possible issues on the roads and plan maintenance strategies.

As shown in Fig. 1, the RMMS we are working on is implemented at four different service layers: (i) the User App, (ii) the Edge Layer, (iii) the Cloud

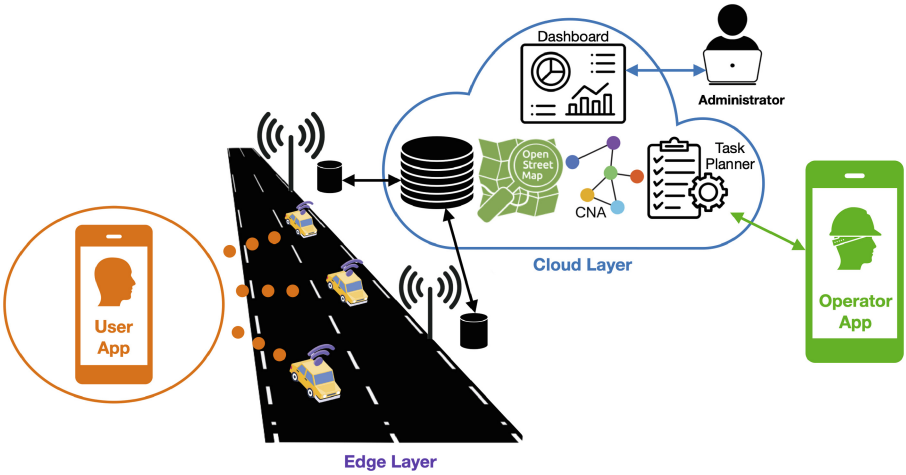


Fig. 1. The proposed architecture for the Road Maintenance Management System

Layer, and (iv) the Operator App. Mobile devices of users travelling along the road are used to collect and process data through several sensors on board the smartphone (e.g., accelerometers, gyroscopes, GPS, camera, and other activated and synchronized sensors). The Edge layer located at the road infrastructure collects and possibly pre-processes data coming from the mobile devices of users and fixed sensors along the smart road. The Cloud layer provides a Big Data oriented storage service and a processing system to analyze all the collected data. CNA tools are available in the Cloud and benefit from both the Big Data stored in the Cloud and the many computational resources. In this way, CNA can implement a complex multi-parametric and deterministic approach to evaluate the effectiveness of connected systems. Data processed by appropriate CNA tools for the Road Anomaly Detection service are then made available to the Administrator through a Cloud dashboard. The Administrator dashboard visualizes on a map the state of deterioration of the road infrastructure using appropriate quality indices, which are estimated on the basis of geolocalized information (e.g., intersections, type of road,...) and data acquired by the various sensors on the smart road and cars.

In our solution, a graph is generated from a specific urban area with intersections as nodes and road segments as edges. This graph can be enriched with the acquired data. For example, specific positions on the road segment collected using the car GPS can become new nodes in the graph; data from an accelerometer can be converted to the International Roughness Index (IRI)[1] and inserted as edge attributes; images can be analyzed to discover road anomalies and map them in the graph. It is also possible to define an "Attention Priority Indica-

[1] https://www.pavement.com.au/news/the-international-roughness-index-iri. Online on October 2024.

tor" (API) [8], useful for identifying the most critical traits and prioritising the scheduling of specific investigations using high-performance survey equipment for the following maintenance planning.

Centrality measures from CNA (e.g., degree or betweenness centrality) can be used to implement the Decision Support System (DSS) to identify the roads with the best and worst quality. Community detection algorithms can identify road quality groups that are considered linkable for the administrator. These measures can be utilized to highlight intersections or roads requiring maintenance on the map, and in a more in-depth analysis we can analyze the variation in their distributions by incorporating road quality indices or attributes into the graph edge weights. Traditional centrality measures typically rely on the local or global structure of the network. However, node-specific factors can also be integrated into network nodes to determine their importance based on intrinsic relevance or external data. This is particularly relevant in urban street networks where several measures have been extended to identify key locations in the urban environment by considering both the network structure and valuable georeferenced socioeconomic data [1,6]. Moreover, complex network disruption can be used as a tool to prioritize roads for maintenance. Node/edge removals can simulate any kind of anomaly (e.g., inaccessible roads for citizens due to traffic accidents, natural disasters as landslides or manhole breaks, double-parked cars, etc.) [9]. The study of the network response to the removal of nodes and/or links can be used by the Administrator to plan road maintenance activities together with link prediction algorithms to identify the roads where an anomaly is most likely to occur.

3 Complex Network Analysis for Smart Road Assessment

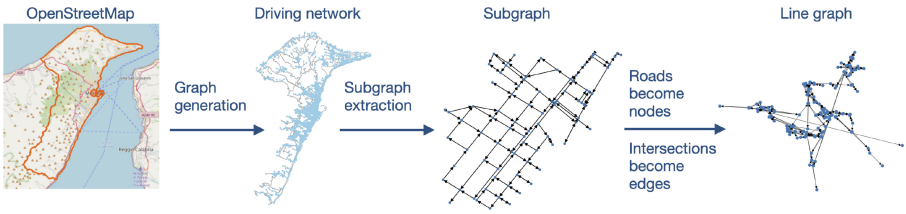


Fig. 2. Road network generation process

This Section describes in detail the design of our CNA solution that can be used to implement the DSS tool in the proposed RMMS. The first step of our solution refers to network generation as described in Fig. 2. The urban road network of a specific city can be built using OSMnx [3], i.e., a Python package which allows the download of geospatial data from OpenStreetMap, work with real-world street networks, and conducts various geospatial analyses. It enables

tasks such as modelling, projecting, visualizing, and analyzing road networks and other geospatial geometries. OSM is a public, worldwide, editable geographical database. It contains points of interest, buildings, addresses, and of course, the road network and its attributes. Once, the OSM road network and its attributes are parsed to GraphML file format, the network file can be converted to the NetworkX format. NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.

OSMnx uses a simplification module that automatically processes network topology from the original raw OSM data such that nodes represent intersections/dead-ends and edges represent the street segments that link them. The most interesting features such as road length, road class (e.g., highway or residential road), number of lanes, or speed limit are the edge-level (i.e., edge attributes). Attributes can also include smart road parameters such as IRI or traffic information or the number of potholes in the road segment. In fact, data associated to the road network will provide additional features to the topological properties of the nodes, and it allows to quantify and qualify the information located in their environments. Node-level attributes are latitude and longitude. Moreover, networks generated with OSMnx can be classified as drivable, walkable, bikeable, or generic street networks. Since our case study involves drivers, the network type is composed of drivable public streets. Being data collected on a small part of the city network, we can create and analyze a subgraph of the whole network using (i) the generated GraphML file specifying a node in OSM node ID and the K-hop neighbor around it, or (ii) the OSMnx package and selecting a specific road (i.e., an address from the considered city) and a distance in meters. K-hop neighbor is one of the fundamental concepts in graph theory. It retrieves the neighbor nodes that a node can reach in K hops (through K edges) in the shortest way. These neighbors are commonly referred to as the k-hop neighbors of the source node. The obtained subgraph G_M models intersections as nodes and road segments as edges, which implies edge-classification tasks in our CNA. Therefore, we can decide to invert the graph (i.e., to create the line graph G_L) so that road segments become nodes and intersections become edges. The line graph of a graph G has a node for each edge in G and an edge joining those nodes if the two edges in G share a common node [11]. For directed graphs, nodes are adjacent exactly when the edges they represent form a directed path of length two. In this way, the most interesting features as road length and road class which are at the edge level become node attributes. We also add a new edge attribute which consists in a list of couples of latitude and longitude and offers the geolocalization of road segments.

3.1 Measures of Centrality

Once we have acquired and prepared the two networks G_M and G_L, we calculate the measures of centrality. In this paper, we analyze six measures of centrality that are briefly described in the following.

Let us consider, a road network as a graph $G = (V, E)$ where V is the set of nodes and E is the set of edges or links.

1) The In-Degree Centrality (DC_I) [10] for a node i is the number of nodes its incoming edges are connected to:

$$DC_I(i) = \sum_{i=1}^{n} A_{i,j} \tag{1}$$

where n is the number of nodes in G and A is the adjacency matrix which represents the connections between nodes in G.

2) The Out-Degree Centrality (DC_O) [10] for a node i is the number of nodes its outgoing edges are connected to:

$$DC_O(i) = \sum_{j=1}^{n} A_{i,j} \tag{2}$$

where n is the number of nodes in G and A is the adjacency matrix which represents the connections between nodes in G. The degree centrality values are normalized by dividing by the maximum possible degree in a simple graph $n-1$ where n is the number of nodes in G.

3) Betweenness Centrality (BC) [7] measures how important the role of a node is in the propagation of information. Some nodes, in fact, act as bridges between different parts of a graph and for this reason they can block the flow of information from one region to another. Specifically, the (shortest-path) betweenness $BC(i)$ of a node i is the sum of the fraction of all-pairs shortest paths that pass through i and is defined as follows:

$$BC(i) = \sum_{j,k \in V} \frac{\sigma(j,k|i)}{\sigma(j,k)} \tag{3}$$

where $\sigma(j,k)$ is the number of shortest paths between an arbitrary pair of nodes j and k, and $\sigma(j,k|i)$ is the number of shortest paths which connect j and k by passing through the node i.

4) Closeness Centrality (CC) [10] measures the proximity between a node and all other nodes in G. The closeness centrality of a node i is the reciprocal of the sum of the shortest path distances from i to all other nodes in G, normalized by $n-1$:

$$CC(i) = \frac{n-1}{\sum\limits_{\substack{j=1 \\ j \neq i}}^{n} d(j,i)} \tag{4}$$

In other words, if the sum of the distances is large, then the closeness is small and vice versa. A node with a high closeness centrality means that it has close relationships with many other nodes.

5) Eigenvector Centrality (EC) computes the centrality for a node using the centrality of its neighbors [4]:

$$EC(i) = \frac{1}{\lambda} \sum_{j \in \mathcal{N}(i)} EC_j = \frac{1}{\lambda} \sum_{j \in N} a_{i,j} EC(j)\,, \tag{5}$$

where $\mathcal{N}(i)$ is the set of neighbors of the node i, N is the set of nodes in G, λ is a constant and $A = (a_{i,j})$ is the adjacency matrix. It can be rewritten as the eigenvector notations as:

$$A\,EC = \lambda\,EC\,. \tag{6}$$

There will be different eigenvalues λ for which a non-zero eigenvector solution exists, and only the greatest one will result in the eigenvector centrality measure.

6) PageRank Centrality (PR) is used by Google to assess the importance of a web page [15]. It expresses the likelihood that a user randomly clicking on a hyperlink will arrive at that particular page. Given a directed graph and its adjacency matrix $A = (a_{i,j})$, the $PR(i)$ of node i is given by:

$$PR(i) = \alpha \sum_j \frac{a_{j,i}}{k_j}\, PR(j) + \beta, \tag{7}$$

where α and β are constants and k_j is the out-degree of node j if such degree is positive, or $k_j = 1$ if the out-degree of j is null. In matrix form, we have:

$$PR = \alpha\, PR\, K^{-1} A + \beta, \tag{8}$$

where β is now a vector whose elements are all equal to a given positive constant and K^{-1} is a diagonal matrix with i-th diagonal element equal to $1/k_i$.

4 Validation

4.1 Experiments in the City of Messina

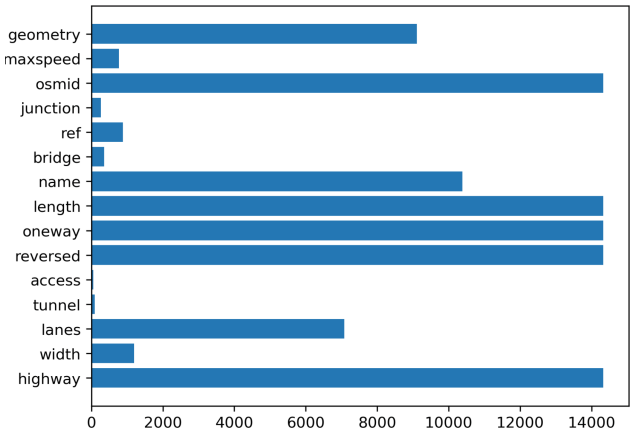


Fig. 3. Edge attribute coverage in OSM Messina Dataset

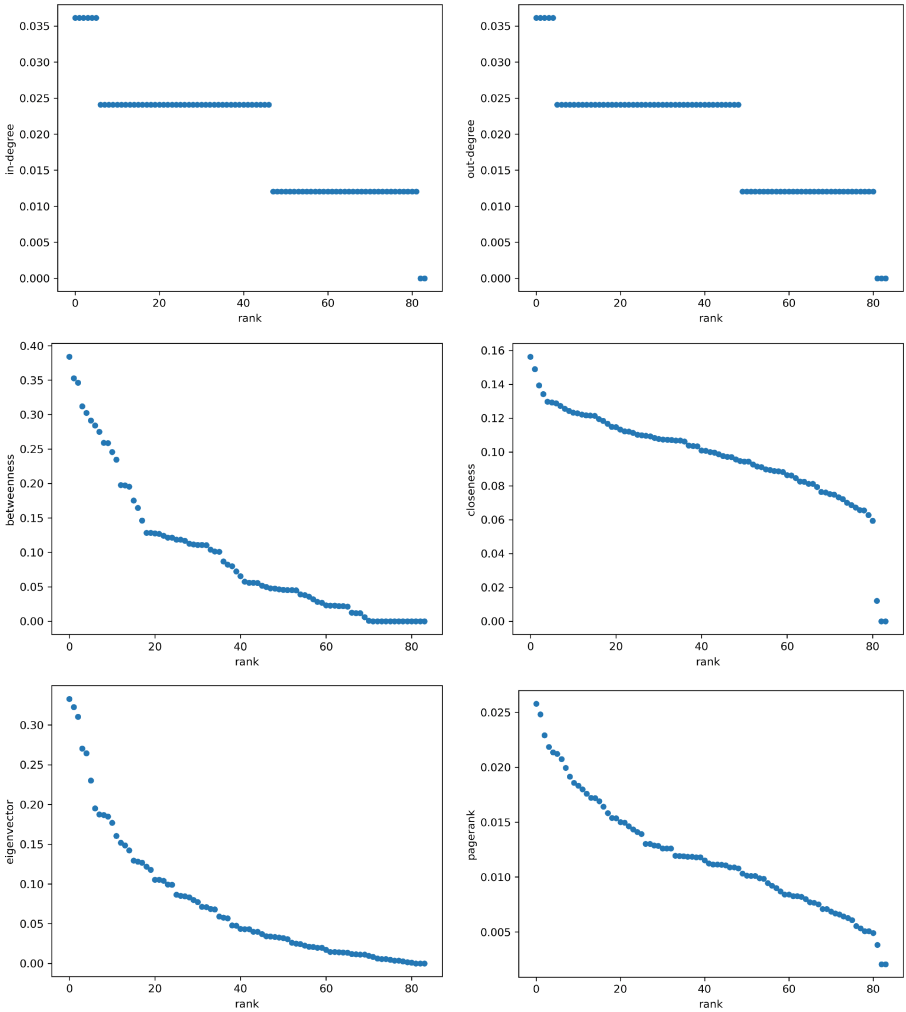


Fig. 4. Centrality distributions in the Messina subgraph (G_M)

Our case study plans a data collection process on specific roads of Messina, Sicily, Italy. Messina is the 3rd-largest city on the island of Sicily, and the 13th-largest city in Italy. The metropolitan city of Messina counts 650,000 inhabitants of which 238,000 live in the actual city. To this purpose, the urban road network of the metropolitan city of Messina is built using OSMnx. Figure 3 shows the attributes available in the Open-StreetMap dataset and their coverage (i.e., the number of edges with the attribute). Once, the OSM road network and its attributes in Messina are parsed to GraphML file format, the network file is converted to the to NetworkX format. The whole Messina network is a directed network with 5,347 nodes and 14,322 edges. Then, two networks are generated:

G_M, which is the Messina subgraph with 84 nodes and 135 edges, and G_L, which is the line graph of G_M with 135 nodes and 226 edges. More specifically, G_M is generated by selecting *Viale Giuseppe La Farina* as address from Messina and a distance of 300 m. Then, we used NetworkX to compute the six centrality measures described in Sect. 3. More specifically, we ranked nodes of G_M and G_L according to their centrality values. Figure 4 shows the centrality distributions in G_M. The x-axis shows the node rank, while the y-axis shows the centrality values. These figures provide information on the most significant nodes. The same experiments were conducted on G_L but are not reported due to space constraints and because no significant differences were found with respect to the results in G_M. These results are still important for the purpose of displaying results on the map.

Fig. 5. Visualization of the most central nodes according to the Betweenness Centrality in the Messina subgraph (G_M)

4.2 Analysis Results

In our experiments, all the used centrality metrics are able to characterise the road's importance in different ways, offering to the system Administrator differ-

Fig. 6. Visualization of the most central nodes according to the Betweenness Centrality in the Messina line graph (G_L)

ent perspectives on the road features. More specifically, DC indicates how many streets are linked at a junction point, suggesting that intersections with a high degree might experience more traffic congestion. BC highlights the importance of intersections in reaching a destination from a starting point. Therefore, we can imagine that in a road network, a high betweenness node is an important intersection that many cars have to cross while taking the shortest path from point A to point B. CC measures how easily a node can reach other nodes, where nodes with high closeness are central and well-connected but may lack shortcut paths. For example, we can imagine that a high closeness node in a road network is an intersection which is extremely central location-wise. EC evaluates the importance of a node based on the prominence of its neighboring nodes. For instance, the value of a retail establishment is influenced not only by its location within the network but also by the significance of nearby businesses [1]. Finally, PR centrality models traffic flow at intersections similarly to how web page redirections work, with the damping factor representing vehicles entering and exiting roads, which helps explain fluctuations in traffic volume [19].

The Administrator should also be able to visualize these results on a map. To this purpose, we used the Python library Plotly to create a scatter plot on a Mapbox map and obtain a visualization which combines the geographical context of a map with the ability to display individual data points as markers. These data points correspond to the nodes in G_M which are geolocalized points to which we have assigned a specific centrality value. Figure 5 shows an example on this kind of visualization on the Administrator dashboard where the BC is chosen to indicate the most central intersections in the road network. Node size and colour are proportional to node betweenness. High betweenness nodes are the bigger and red ones. Then, we used the Python library Folium to create the map and add the geolocalized lines corresponding to the nodes in G_L. Figure 6 shows the visualization of the most central streets in the road network according to their BC values. In this case, a colour map is used to show the different centrality values. High betweenness nodes are coloured dark blue.

5 Conclusion

In this paper, we proposed an innovative solution for smart road network-scale asset management, based on CNA to support road administrators in asset network evaluation and classification. We also validated the potential of our CNA-based methodology on a real urban road network generated from the city of Messina (Italy). This was useful to verify the potential applicability of centrality measures from CNA to road network analysis for management aims. The obtained results were interesting and evidenced that centrality metrics are able to capture the importance of intersections and road segments in the proposed street network offering the road administrators preliminary knowledge on places to pay more attention to when planning maintenance activities as they may experience more traffic congestions or prevent drivers from reaching a destination. Despite our analysis relying only on the network topology, in the next steps of this research, specific attributes of the road segments (related for example to quality condition, defects, traffic, functional classification, etc.) will be involved to deepen the analysis and provide more reliable rankings and evaluations.

Acknowledgements. This work has been partially supported by the European Union (NextGeneration EU), through the MUR-PNRR project SAMOTHRACE (ECS00000022).

References

1. Agryzkov, T., Tortosa, L., Vicent, J.F., Wilson, R.: A centrality measure for urban networks based on the eigenvector centrality concept. Environ. Plan. B: Urban Anal. City Sci. **46**(4), 668–689 (2019)
2. Ahmadzai, F., Rao, K., Ulfat, S.: Assessment and modelling of urban road networks using Integrated Graph of Natural Road Network (a GIS-based approach). J. Urban Manag. **8**(1), 109–125 (2019)

3. Boeing, G.: OSMNX: new methods for acquiring, constructing, analyzing, and visualizing complex street networks. Comput. Environ. Urban Syst. **65**, 126–139 (2017)
4. Bonacich, P.: Power and centrality: a family of measures. Am. J. Sociol. **92**(5), 1170–1182 (1987)
5. Bosurgi, G., Bruneo, D., De Vita, F., Pellegrino, O., Sollazzo, G.: A web platform for the management of road survey and maintenance information: a preliminary step towards smart road management systems. Struct. Control. Health Monit. **29**(3), e2905 (2022)
6. Bowater, D., Stefanakis, E.: Extending the Adapted PageRank Algorithm centrality model for urban street networks using non-local random walks. Appl. Math. Comput. **446**, 127, 888 (2023)
7. Brandes, U.: On variants of shortest-path betweenness centrality and their generic computation. Soc. Netw. **30**(2), 136–145 (2008)
8. Ficara, A., et al.: Distributed services for pavement anomaly detection in smart roads. In: IEEE International Symposium on Measurements and Networking (M&N), pp. 1–6 (2024)
9. Ficara, A., Fiumara, G., Fazio, M., Celesti, A., Villari, M.: Novel strategies for road network disruption analysis. In: Proceedings of the IEEE/ACM 16th International Conference on Utility and Cloud Computing, UCC '23. Association for Computing Machinery, New York, NY, USA (2024)
10. Freeman, L.C.: Centrality in social networks conceptual clarification. Soc. Netw. **1**(3), 215–239 (1978)
11. Harary, F., Norman, R.Z.: Some properties of line digraphs. Rendiconti del Circolo Matematico di Palermo **9**(2), 161–168 (1960)
12. HOLME, P.: Congestion and centrality in traffic flow on complex networks. Adv. Complex Syst. **06**(02), 163–176 (2003)
13. Jayaweera, I.: Centrality measures to identify traffic congestion on road networks: a case study of Sri Lanka. IOSR J. Math. **13**(02), 13–19 (2017)
14. Kirkley, A., Barbosa, H., Barthelemy, M., Ghoshal, G.: From the betweenness centrality in street networks to structural invariants in random planar graphs. Nat. Commun. **9**(1), 2501 (2018)
15. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. Technical report 1999-66, Stanford InfoLab (1999)
16. Park, K., Yilmaz, A.: A social network analysis approach to analyze road networks. Am. Soc. Photogrammetry and Remote Sensing Annual Conference 2010: Opportunities for Emerging Geospatial Technologies, vol. 2, pp. 754–759 (2010)
17. Pompigna, A., Mauro, R.: Smart roads: a state of the art of highways innovations in the smart age. Eng. Sci. Technol. Int. J. **25**, 100, 986 (2022)
18. Pung, J., D'Souza, R.M., Ghosal, D., Zhang, M.: A road network simplification algorithm that preserves topological properties. Appl. Netw. Sci. **7**(1) (2022)
19. Zhang, T., Li, G., Xu, Y., Yang, Y., Lv, S.: Prediction of transportation network based on pagerank algorithm. In: Proceedings of the 2016 5th International Conference on Advanced Materials and Computer Science, pp. 88–94. Atlantis Press (2016)

Author Index

A
Aashish 237
Addai, Emmanuel 100
Agarwal, Nitin 100, 113
Agrawal, Abhigya 211
Aida, Masaki 187
Akgun, Mehmet Burak 292
Alahmadi, Sarah 87
Ali, Hesham H. 304
AlMahmoud, Hind 152
Almulhim, Aljohara 343
Anik, Md Saidul Hoque 211
Anlauf, Tomas 3
Aoyagi, Haruto 402
Azad, Ariful 211

B
Barbour, Jason 224
Bosurgi, Gaetano 447
Brambilla, Marco 66
Brandt-Tumescheit, Fabian 139
Brede, Markus 87
Bulut, Mehmet Eren 292

C
Calcini, Niccolò 39
Cau, Erica 166
Cena, Federica 435
Chattanachot, Supharoek 389
Cherifi, Hocine 53, 224, 250, 279
Colaiori, Francesca 66
Cruz-Bonilla, Erika 279

D
Dagtas, Selimhan 113
Daijavad, Shayan 328
Diallo, Cherif 53
Diop, Issa Moussa 53, 279
Diwakar, Parul 237
Durandard, Noé 356

E
Enright, Jessica 76

F
Failla, Andrea 166
Fazio, Maria 447
Ficara, Annamaria 447
Fushimi, Takayasu 402

G
Gerharz, Frieda 139
Gharibian, Arootin 15
Guarino, Stefano 66
Guinand, Frédéric 389

H
Hasan, Mohammad Al 343
Head, Michael 87
Hernández-Rosales, Maribel 279
Hoyle, Rebecca 87
Hunter, Ethan 76

I
Iguchi, Taku 316
Ishibashi, Keisuke 196

J
Janca, Ondrej 15
Jolivet, Renaud B. 39

K
Kathuria, Hritika 369
Kawato, Kiri 187
Knežević, Dušica 374
Kobayashi, Sachio 415
Kotyada, Venkata Narendra 237
Koyama, Kazuki 127
Kriegova, Eva 3, 15, 27

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2025
H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2024, SCI 1190, pp. 459–460, 2025.
https://doi.org/10.1007/978-3-031-82439-5

Kubikova, Kristyna 3
Kudelka, Milos 3, 15, 27
Kyzym, Vitalii 39

L
Lai, Mirko 435
Lavangnananda, Kittichai 389

M
Mallmann-Trenn, Frederik 152
Martínez Márquez, Rafael Alejandro 267
Mazza, Fabio 66
Meloni, Sandro 66
Menezes, Telmo 356
Meyerhenke, Henning 139
Migler, Theresa 328
Miyagi, Shigeyuki 316

N
Neelima, N. V. 425

O
Ochodkova, Eliska 3, 15, 27
Ohnishi, Takaaki 127
Olmez, Sedar 415
Oryu, Tatsuya 196

P
Papik, Lukas 27
Patanè, Giuseppe 267
Piccardi, Carlo 66
Pierri, Francesco 66
Pramanik, Soumajit 237

R
Radovanović, Miloš 374
Rossetti, Giulio 166
Roth, Camille 356
Ruffo, Giancarlo 435

S
Sakaguchi, Masanori 127
Sakai, Osamu 316
Sakiyama, Tomoko 177
Saracco, Fabio 66
Savić, Miloš 374
Seba, Hamida 224, 250
Shriyam, Shaurya 425
Singh, Rishi Ranjan 237
Siu, Ka-Chun 304
Sollazzo, Giuseppe 447

T
Takahara, Akihiro 177
Temiz, Afra 292
Thelagathoti, Rama Krishna 304
Togni, Olivier 250
Tupikina, Liubov 369

U
Uchimura, Kei 316

V
Vilella, Salvatore 435

W
Wei, Yongping 425
Wolff, Christian 279
Wu, Shuanglei 425

Y
Yamamoto, Tetsuya 187
Yassin, Ali 224, 250
Yousefi, Niloofar 100, 113