



Politechnika Wrocławska

FIELD OF SCIENCE: Engineering and Technology

DISCIPLINE OF SCIENCE: Information and Communication Technology

DOCTORAL DISSERTATION

On Spreading Phenomena and Influence Maximisation in Multilayer Networks

mgr inż. Michał Jerzy Czuba

Supervisor:

dr hab. inż. Piotr Bródka, prof. Politechniki Wrocławskiej

Assistant supervisor:

dr inż. Katarzyna Musiał-Gabryś, prof. University of Technology Sydney

Keywords: complex networks, graph mining, influence maximisation, multilayer networks, spread control

WROCŁAW 2026



Albrecht Dürer, "*Melencolia I*", 1514 (public domain).

Acknowledgments

This work could not have been completed without the support of people who, although not listed as authors, stood behind me and provided the conditions necessary for this research to be carried out.

I am particularly indebted to the academics who supervised my work, guided me through the investigation of this challenging topic, and devoted countless hours to discussions and constructive criticism: Piotr Bródka and Katarzyna Musiał-Gabryś. I am sincerely grateful for your patience and for giving me the opportunity to engage globally with the academic community — from the soil of Lower Silesia to the world's antipodes.

I would also like to acknowledge members of my family, foremost those who died during the COVID-19 pandemic, as this tragedy inspired my deeper interest in the study of spreading phenomena in complex systems. Finally, I extend my sincere gratitude to my closest family — particularly my Mother and Father — and to my friends, whose unwavering support helped me persevere in pursuing this endeavour. Thank you all!

Abstract

Networks offer a convenient framework for representing many complex systems. Social bonds between individuals, interactions among proteins in living organisms, transportation routes, or water supply systems: all of these can be modelled using graph-based formalisms. Over the years, numerous approaches to network modelling have been developed. In particular, the notion of multilayer networks has become successful, as it enables the simultaneous representation of multiple types of relations. Conceptually, a multilayer network can be seen as a stacked collection of graphs, where each one is spanned on the same node set (or on a subset thereof). Contextually, this structure can be obtained, e.g., from users aggregated across several social platforms together with the friendship relations between them.

One of the most important aspects of networks concerns their dynamics. This may take various forms, including structural changes in topology or fluctuations in the states of agents. The latter has typically been examined within the framework of spreading processes and underpins many domain-specific research areas such as epidemic modelling, opinion dynamics, and information diffusion. A key issue in this context is how to propagate a given phenomenon across a network as effectively as possible. This is traditionally referred to as the influence maximisation problem, which has a broad practical impact. In its canonical formulation, it seeks to identify a set of agents that, when chosen to initiate the spread, maximise the eventual reach of the diffusion process. Over the years, however, this problem has been reformulated in numerous ways, and various methodological paradigms have been leveraged to approach it.

This doctoral dissertation aims to address the problem of modelling spreading phenomena in multilayer networks and to analyse various approaches to solving the influence maximisation problem in such structures. As this work has been carried out within the domain of Information and Communication Technology, a complementary challenge was undertaken: to contribute to reliable and open research by publishing data, algorithms, methods, and code alongside the scientific findings. Seven research gaps have been identified within the scope of this dissertation. These gaps were subsequently transformed into corresponding objectives and prerequisites leading towards the completion of the overarching goal. The goal itself has been formulated as to *develop and evaluate approaches for maximising influence in multilayer networks by leveraging selected paradigms that provide complementary perspectives on complex systems: a notion of actor centrality, controllability means, graph-based machine learning, and structural interventions on the analysed system*. The research presented here approaches

this agenda through a collection of seven (P1-P7), thematically related publications, together with the binding part that demonstrates how each of them contributes to the problem under consideration.

The first research prerequisite: *identify various methodologies for modelling spreading phenomena in multilayer networks, and select a single mechanism under which all research objectives will be evaluated* has been addressed through the development of a framework (**network-diffusion**) that represents any configuration of coexisting phenomena by means of an n -dimensional grid. Furthermore, a recently established approach to modelling actor-based spread — protocol functions that govern how layer-level activations are aggregated — has been adapted to the multilayer setting and extended from the Linear Threshold Model to its counterpart, the Independent Cascade Model.

The subsequent one — *to develop metrics for quantifying spreading effectiveness in multilayer networks, enabling comparison of spreads triggered under varying conditions* — has resulted in a set of new measures designed to assess influence maximisation methods from both static and dynamic perspectives. In addition, an evaluation approach capable of abstracting over specific spreading conditions has been proposed. Finally, a methodology for estimating the spreading capabilities of individual actors has been coined and employed accordingly.

The third prerequisite was formulated as *to provide multirelational data that can serve as a structural backbone for experiments on influence maximisation and develop methods capable of real-time augmentation*. It has been addressed by acquiring and publishing a new longitudinal dataset that captures the physical interactions within an isolated group of people. A representative collection of networks was also compiled and used throughout all simulations covering this doctoral research, thereby ensuring coherence. Moreover, a novel synthetic generator of power-law multilayer networks, **mABCD**, has been introduced, distinguished by its efficiency and its ability to model nuanced interlayer dependencies.

After implementing the three prerequisites, works aimed at investigating various perspectives on maximising influence were carried out. Within the following objective: *evaluate various rank refining methods for influence maximisation in multilayer networks and assess how they differ in terms of spreading effectiveness*, a comprehensive study of the most basic methods considered in this research was conducted. As a result, sixteen heuristics were evaluated, amongst them three (based on centrality metrics originally defined for monoplex networks) were extended to the multilayer scenario. Moreover, a new approach, *Neighbourhood Size Discount*, was proposed. The main conclusion was that all evaluated parameters: those pertaining to the spreading model, the network type, the seed budget, and the seed selection method, collectively determine the overall effectiveness of influence maximisation.

The consecutive objective was formulated as: *propose a seed-selection approach that leverages the minimal dominating set for the influence maximisation problem in multilayer networks and verify its effectiveness*. The work addressing this objective resulted in extending the concept of the minimal dominating set — a collection of agents through which one can

control the entire system — to the actor-based standpoint. It was subsequently used to enhance the most effective rank-refining seeding methods to test whether triggering diffusion from the dominating set leads to greater network coverage. The results revealed that this method is significantly advantageous, although only under particular spreading conditions.

Another paradigm employed was machine learning. The study concerning this objective aimed to *develop an inductive graph neural network that predicts which actors in a given multilayer network are the most effective spreading seeds, and validate its performance against other state-of-the-art methods*. To achieve this, a dataset comprising the spreading potential of every actor within more than two hundred multilayer networks was created. Then, a graph neural network, *TopSpreadersNetwork*, was developed to perform the task of predicting these values. The main design goal was to make it capable of inferring from unseen network data, a feature uncommon in the domain of machine learning on graphs. The comparative study showed it is able to generalise and yields better results than competing methods.

The final objective realised within the scope of this dissertation was to *design a method to manipulate the structure of a multilayer network at a macro scale, enabling comparison of spreading effectiveness between the original and modified networks, and identify the structural properties that facilitate the influence maximisation process*. To this end, a conceptual study examining how the parameters of the **mABCD** model affect the network’s spreading capabilities was performed. Moreover, the research resulted in a method to embed a real network into the configuration space of **mABCD**, which may be particularly relevant in the context of digital twins of complex networked systems. The results demonstrate, i.e., a strong impact of the intercommunity noise level on spreading effectiveness.

In conclusion, this dissertation presents multiple perspectives on the influence maximisation problem in multilayer networks. It also addresses several preliminary gaps, such as the modelling of spreading phenomena, whose resolution enables subsequent domain-specific studies. The results demonstrate that the problem in the considered network type exhibits greater complexity than its canonical form. The contributions of this work encompass advances in multilayer network research, including novel spreading and centrality metrics, new modelling approaches, methods for identifying the most influential actors, algorithms, datasets, and a synthetic graph generator. The central goal has been realised, albeit with some caveats arising from the limitations of the proposed methods.

Streszczenie

Sieci stanowią przystępną metodę reprezentacji wielu systemów złożonych. Więzi społeczne, interakcje pomiędzy białkami, szlaki transportowe czy też systemy zaopatrzenia w wodę — wszystkie one mogą być modelowane przy użyciu formalizmów bazujących na grafach. To właśnie między innymi dlatego w ostatnich latach opracowano liczne metody modelowania tego typu struktur. W szczególności, dużą popularność zyskały sieci wielowarstwowe, które umożliwiają jednoczesną reprezentację wielu typów relacji. Koncepcyjnie, sieć wielowarstwową można postrzegać jako zbiór grafów, z których każdy rozpięty jest na tym samym zbiorze węzłów (lub jego podzbiorze), a same węzły stanowią reprezentację aktorów systemu względem danej relacji. Taka struktura może zostać uzyskana np. poprzez agregację użytkowników z kilku portali społecznościowych wraz z relacjami znajomości pomiędzy nimi.

Jednym z najważniejszych aspektów sieci jest ich dynamika. Może ona przyjmować różne formy, obejmujące zarówno zmiany strukturalne, jak i fluktuacje stanów aktorów budujących ją. Ten drugi rodzaj zjawisk badany jest zazwyczaj w ramach procesów rozprzestrzeniania i stanowi podstawę wielu obszarów badań specyficznych dla danych dziedziny, jak np: modelowanie epidemii, dynamika opinii czy dyfuzja informacji. Kluczowym zagadnieniem w tym kontekście jest znalezienie sposobu na możliwie najefektywniejszą propagację danego zjawiska w sieci (w wymiarze praktycznym może być to reklama, opinia, wirus itp.). Problem ten tradycyjnie rozpatrywany jest jako zadanie maksymalizacji wpływu. W swojej kanonicznej postaci polega ono na identyfikacji takiego zbioru agentów, który jeśli użyty do zainicjowania propagacji, zapewni jej maksymalny zasięg. Z biegiem lat problem ten bywał jednak wielokrotnie przeformułowywany, a do jego rozwiązania wykorzystywano różne paradygmaty.

Celem niniejszej rozprawy jest ujęcie problemu modelowania zjawisk rozprzestrzeniania w sieciach wielowarstwowych oraz analiza różnych podejść do rozwiązywania zadania maksymalizacji wpływu w tych strukturach. Z uwagi na to, że praca została zrealizowana w dyscyplinie Informatyki Technicznej i Telekomunikacji, jej równoległym wynikiem są opublikowane zbiory danych, algorytmy, metody, ich implementacje oraz środowiska symulacyjne. W ramach rozprawy zidentyfikowano siedem luk badawczych, a te następnie przekształcono w odpowiadające im cele oraz prerekwizyty, których uprzednie wykonanie pozwala na realizację celów. Sam cel główny natomiast sformułowano jako *opracowanie i ewaluację podejść do maksymalizacji wpływu w sieciach wielowarstwowych poprzez wykorzystanie wybranych paradygmatów dostarczających komplementarne perspektywy na systemy złożone: pojęcia*

centralności w sieci, kontroli/sterowania w sieciach, uczenia maszynowego na grafach oraz strukturalnych interwencji w analizowanym systemie. Przedstawione badania realizują ten cel poprzez zbiór siedmiu tematycznie powiązanych publikacji wraz ze wstępem merytorycznym, ukazującym wkład każdej z nich w rozpatrywany problem.

Pierwszy prerekwizyt: *identyfikacja metod modelowania zjawisk rozprzestrzeniania w sieciach wielowarstwowych oraz wybór jednego mechanizmu, w oparciu o który zrealizowane zostaną wszystkie cele badawcze*, osiągnięto poprzez opracowanie środowiska symulacyjnego (**network-diffusion**), które pozwala na reprezentowanie dowolnej konfiguracji współlistniejących zjawisk za pomocą n -wymiarowej siatki stanów. Ponadto, niedawno zaproponowane podejście do modelowania dyfuzji w sieci z perspektywy aktora (tj. funkcje protokołów określające sposób agregacji aktywacji uzyskanych na poziomie poszczególnych warstw) zostało zaadaptowane do tematu pracy i rozszerzone z modelu prognozy liniowego na model niezależnych kaskad.

Drugie z wymagań wstępnych — *opracowanie miar opisujących efektywność rozprzestrzeniania w sieciach wielowarstwowych, umożliwiającich porównanie dyfuzji inicjowanych w różnych warunkach* — doprowadziło (zgodnie ze swym brzmieniem) do zaproponowania zestawu nowych metryk pozwalających na porównywanie dyfuzji zarówno z perspektywy statycznej, jak i dynamicznej. Opublikowano również autorską metodę ewaluacji pozwalającą abstrahować od konkretnych warunków rozprzestrzeniania. Ponadto sformułowano i przebadano metodę estymacji zdolności do propagacji pojedynczych aktorów.

Trzeci prerekwizyt określono jako: *pozyskanie wielorelacyjnych danych mogących stanowić strukturalne zaplecze eksperymentów nad maksymalizacją wpływu oraz opracowanie metod umożliwiających ich uzupełnianie w czasie rzeczywistym.* Został on zrealizowany poprzez opublikowanie nowej sieci opisującej fizyczne interakcje w izolowanej grupie osób. Reprezentatywny zbiór danych wykorzystanych we wszystkich symulacjach zapewnił natomiast spójność badań. Ponadto, zaproponowano nowy syntetyczny generator sieci wielowarstwowych o rozkładzie potęgowym, **mABCD**, wyróżniający się wysoką wydajnością oraz zdolnością do modelowania złożonych zależności pomiędzy warstwami.

Po zrealizowaniu prerekwizytów przeprowadzono prace ukierunkowane na osiągnięcie głównego celu rozprawy. W ramach *ewaluacji metod rankingowych w zadaniu maksymalizacji wpływu w sieciach wielowarstwowych oraz ocenie różnic w ich skuteczności* wykonano kompleksowe badanie najbardziej podstawowego podejścia z rozpatrywanych. Oceniono szesnaście heurystyk, w tym trzy rozszerzone do scenariusza wielowarstwowego na bazie centralności pierwotnie zdefiniowanych dla sieci jednowarstwowych. Zaproponowano również nową metodę: *Neighbourhood Size Discount*. Główny wniosek wskazuje, że wszystkie analizowane parametry, zarówno te dotyczące modelu rozprzestrzeniania, typu sieci, wielkości zbioru początkowego oraz metody jego wyboru, łącznie determinują ogólną efektywność maksymalizacji wpływu.

Kolejny cel stanowiło *zaproponowanie metody wyboru aktorów początkowych wykorzystującej minimalny zbiór dominujący w zadaniu maksymalizacji wpływu w sieciach wielowarstwowych oraz weryfikację jej skuteczności.* Realizacja tego celu doprowadziła do rozszerzenia pojęcia

minimalnego zbioru dominującego — zbioru agentów umożliwiających kontrolę nad całym systemem — na sieci wielowarstwowe. Następnie, z wykorzystaniem najskuteczniejszych metod rankingowych, sprawdzono czy inicjowanie rozprzestrzeniania z poziomu zbioru dominującego zwiększa jego zasięg. Wyniki wykazały znaczną, lecz ograniczoną do specyficznych warunków, przewagę tego podejścia.

Badania w obszarze uczenia maszynowego miały na celu *opracowanie indukcyjnej grafowej sieci neuronowej do predykcji najefektywniejszych inicjatorów dyfuzji w danej sieci wielowarstwowej oraz walidację jej działania w porównaniu z konkurencyjnymi podejściami*. W tym celu utworzono zbiór danych ponad dwustu sieci wielowarstwowych z etykietami określającymi potencjał rozprzestrzeniania każdego aktora. Następnie opracowano grafową sieć neuronową *TopSpreadersNetwork*, zdolną do przewidywania tych wartości. Kluczowym założeniem projektowym było umożliwienie wnioskowania na podstawie wcześniej niewidzianych danych, co stanowi rzadkość w obszarze rozpatrywanego problemu. Badania porównawcze wykazały zdolność opracowanej metody do generalizacji oraz przewagę nad rozwiązaniami konkurencyjnymi.

Ostatnie zadanie polegało na *zaprojektowaniu metody manipulowania strukturą sieci wielowarstwowej w skali makro, umożliwiającej porównanie efektywności rozprzestrzeniania pomiędzy siecią oryginalną i zmodyfikowaną oraz identyfikację własności strukturalnych sprzyjających maksymalizacji wpływu*. W tym celu przeprowadzono studium koncepcyjne analizując wpływ parametrów modelu *mABCD* na możliwości propagacji w sieci. Opracowano również metodę osadzania rzeczywistej sieci w przestrzeni konfiguracji *mABCD*, co może mieć szczególną wartość w kontekście tzw. *cyfrowych bliźniaków* systemów złożonych. Uzyskane wyniki wskazują między innymi na silny wpływ poziomu szumu międzyspołecznościowego na efektywność rozprzestrzeniania.

Podsumowując, rozprawa przedstawia cztery perspektywy dotyczące problemu maksymalizacji wpływu w sieciach wielowarstwowych. Odnosi się również do szeregu prerokwizytów (takich jak modelowanie zjawisk rozprzestrzeniania) których rozwiązanie umożliwiło przeprowadzenie właściwych badań. Uzyskane rezultaty pokazują, że problem w analizowanym typie sieci cechuje się większą złożonością niż jego kanoniczna postać. Wkład pracy obejmuje rozwój badań nad sieciami wielowarstwowymi, w tym nowe miary oceny rozprzestrzeniania, centralności, podejścia do modelowania procesów rozprzestrzeniania, metody identyfikacji najbardziej wpływowych aktorów, zbiory danych oraz syntetyczny generator grafów. Nadrzędny cel został osiągnięty, choć z zastrzeżeniami wynikającymi z ograniczeń zaproponowanych metod.

Contents

Abstract	i
Streszczenie	v
Contents	1
1 Introduction	3
1.1 Complex systems and underlying dynamical phenomena	4
1.2 Key concepts	6
1.3 Challenges within the field	12
1.4 Identified research gaps	13
1.5 Innovation of the research	22
2 Research Goal	25
2.1 The goal, resulting objectives, and prerequisites	25
2.2 List of publications	26
2.3 Addressing the research items	27
3 Conclusions	43
4 Publications	47
4.1 Simulating Spreading of Multiple Interacting Processes in Complex Networks	48
4.2 Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks	59
4.3 Peer Interaction Dynamics and Second Language Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social Network Analysis	79
4.4 Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model	139
4.5 Applicability of the Minimal Dominating Set for Influence Maximization in Multilayer Networks	165
4.6 Identifying Super Spreaders in Multilayer Networks	189
4.7 Multilayer Artificial Benchmark for Community Detection (mABCD)	201
A Scientific Achievements	223
A.1 Research Projects	223

A.2 Collaborators and Advisors	223
A.3 Research Internships	224
A.4 Scientific Conferences	225
A.5 Journal and Conference Reviews	226
A.6 Other Scientific Activities	226
Bibliography	229

Chapter 1

Introduction

Even before the emergence of network science, whether as an independent discipline or as a branch of Information and Communication Technology, the idea of modelling complex systems as graphs had already attracted attention from researchers across various domains. This doctoral dissertation stands in that long series of studies addressing problems within the realm of networks. However, unlike many attempts made over recent decades, both the research problem and the methodology adopted here are formulated using the apparatus provided by computer science. In particular, the thesis focuses on spreading phenomena and influence maximisation in multilayer networks. The text presented below should be regarded as a keystone in the arch formed by a series of publications addressing it through different perspectives. Its primary purpose is thus to clearly articulate the research goal, shed light on its specific manifestations (as explored in the accompanying papers), and present the conclusions that emerge from them.

This dissertation is organised into four main parts. The first one (Chp. 1) outlines the problems relevant to the research domain. In particular, Sec. 1.1 provides a general overview of the field, while Sec. 1.2 introduces three key concepts necessary for understanding the conducted research. The major challenges in the field, with a particular focus on the aforementioned concepts, are discussed in Sec. 1.3, and subsequently formulated as the research gaps that this dissertation aims to address (Sec. 1.4). The second part of the thesis (Chp. 2) presents how the main research problem has been formulated and solved. Sec. 2.1 defines the overarching goal, along with the specific objectives and prerequisites. The list of publications comprising this dissertation is provided in Sec. 2.2, followed by a discussion in Sec. 2.3 that explains how each article contributes to achieving a particular objective or prerequisite. The third part (Chp. 3) presents the conclusions drawn from the doctoral research regarding the stated goal. Finally, since this dissertation is composed as a collection of research papers: **P-1–P-7**, their full versions, as published, are included in Chp. 4. The dissertation is supplemented with an appendix (App. A) that concisely summarises the scientific achievements of the author obtained during the doctoral education.

For the sake of clarity, it is important to note that the coherence of mathematical notation is maintained throughout the text summarising the conducted research. However, despite the author's sincere efforts, the articles included in Chp. 4 may employ different symbols to represent the same concepts introduced below. This variation arises from the conventions adopted in the distinct research domains corresponding to the publication venues (e.g., notation in linguistics differs from that used in mathematics), as well as from the time span between the works, during which the author gained greater proficiency in the use of mathematical formalism. Nevertheless, each article comprising this dissertation is internally consistent and should primarily be regarded as an independent piece of research, which can then be cross-referenced with this summarising text.

1.1 Complex systems and underlying dynamical phenomena

Many real-world complex systems can be represented as networks. Friendship relations on social platforms, organisational hierarchies within corporations, interactions between proteins governing biological processes, or transportation routes for goods, data, and raw materials: all these structures can be modelled within a unified conceptual framework of nodes and links between them. Over the years, numerous approaches to describing complex systems have been developed. Initially, such structures were modelled either as small graphs, where each node and edge was introduced by deliberate decision of a supervising researcher, or as stochastic structures generated according to mathematical formulations that globally approximate the given system. However, this situation changed about two decades ago, when the availability of big data and increased power of computers enabled the modelling of far more extensive structures, which in turn contributed to the growth of complex systems to a previously unseen scale. This breakthrough gave rise to a new discipline: network science [1]. Initially shaped primarily by physicists, mathematicians, and sociologists, the field soon became an important branch of computer science due to its ability to employ large-scale experimental and computational methods. Consequently, several subfields emerged, such as computational social science [2] and computational network science [3].

The rise of computational methods has also left a mark on network models. Their evolution over recent years traces attempts to characterise various, usually more detailed, nuances of complex systems. Three main paradigms can be observed in this regard. On one hand, temporal aspects, since most systems evolve over time, as shall the topology of the network that illustrates them. Then, the way in which relationships are described also plays a significant role: links can express different types of connection, directionality of interaction, and may occur in cardinalities exceeding the classical pairwise notation (thus so-called hypergraphs [4]). Finally, the attributes of the underlying graph are also often considered, such as node types [5] or connection strengths. Each of these paradigms gives rise to its own family of models: for instance, temporality in networks can be depicted using time windows [6] or interaction (event) streams [7]. At this point it is good to note that the proliferation of network models reveals

the richness of the field and the broad range of problems it is capable to address.

Multilayer networks, which, as noted above, constitute a leitmotif of this research, are one of many such models. Intuitively, it is not difficult to imagine that we, humans live surrounded by many different kinds of relationships. Some of these are close and personal, such as friendship ties or even innate as family bonds, while others are weaker and maintained as a form of obligation, for instance professional connections. The same applies to the virtual world: many of us communicate through social media platforms or follow people on Facebook we have never met in person. In order to fully capture this diversity of relationships in complex (not only social) systems, multilayer networks [8] have been successfully introduced. In such structures, different relations are represented as layers. Each layer is, in fact, a classical graph in which nodes represent entities and links represent one specific type of tie. Multilayer networks have proved particularly successful in the study of social systems [9], but they have also gained recognition in modelling other types of systems, such as transportation [10], physiological [11], and economic ones [12]. Despite more than decade has passed since formal introduction of this concept, it still remains a subject of extensive research [13], [14].

While networks provide a powerful apparatus for representing complex systems, an equally important aspect lies in their dynamics. This may take various forms, such as structural changes or fluctuations in the states of the entities that constitute the system. The latter has typically been studied within the framework of spreading processes and forms the backbone of many domain-specific fields, such as epidemic modelling, opinion dynamics, and information diffusion in society. Much like the development of graph theory, this line of research has a history that predates the formal emergence of network science.

Epidemiology was the field in which the first attempts to understand the mechanisms underlying disease propagation were made, leading to the development of the earliest spreading models, such as the Susceptible-Infected-Susceptible and Susceptible-Infected-Recovered [15]. The predominant reason for this early focus was that epidemic propagation is one of the most observable spreading processes. Contagious diseases have been an inherent part of humankind since the dawn of history, leaving a profound mark on societies. As the recent outbreak of the COVID-19 pandemic has shown, epidemics are not relics of the past gazing at us from scarred paintings, sorrowful literature, or mass graveyards: they accompany us continuously and pose a constant threat to the fragile balance of the modern world. Therefore, such phenomena were the first to attract the attention of researchers, prompting efforts to understand and control how contagions propagate through populations. The ongoing reports of new outbreaks confirm that this line of research remains far from concluded [16]. Consequently, a substantial body of work has been devoted to understanding the mechanisms of disease propagation, accompanied by the development of computational approaches in this domain [17], [18], [19]. Epidemiology, although a primary area in which spreading phenomena have been examined, is not the only one. In recent years, the rapid expansion of the Internet and the resulting availability of big data have enabled researchers to investigate the formation of opinions [20], diffusion of rumours [21], and (dis)information [22], particularly in the context of hybrid

war between the Russian Federation and NATO [23]. Cybersecurity has likewise emerged as a field of considerable interest, focusing on the modelling of how computer viruses and malware spread across digital infrastructures [24]. Beyond that, diffusion processes have been extensively explored in infrastructure networks [25], financial domain (for instance in analyses of cascade effects triggered by economic shocks [26]) or in the area of medical genetics (for instance to improve replication of genetic studies by using spreading process over the protein network [27]).

The topic of dynamics in complex systems, when translated into the study of spreading processes in networks, naturally raises a subsequent question: if such phenomena can be observed, can they also be controlled? And if so, what outcomes are to be pursued and what means can be employed to achieve them? Two opposing objectives can be distinguished in this context: maximising or minimising the spread (with their derivatives such as accelerating or slowing down the dynamics). The former is typically examined within the framework of influence maximisation [28], while the latter corresponds to influence suppression or network immunisation [29]. Ultimately, both directions can be viewed as attempts to steer the system (represented by a network) from one state to another or to employ certain mechanisms that preserve it in its current configuration. Hence, these problems can be also framed within the broader principles of controlling complex systems [30].

Considering the variety of methodologies proposed in the literature to control spreading processes, most studies focus on changing the final number of affected nodes under "use cases" like informing, infecting, or convincing them. The means employed to achieve these goals cover a broad range of techniques, such as seed selection strategies [31], [32], which aim to identify the most effective set of nodes from which diffusion should be initiated, usually under minimal cost constraints. Another family of approaches extends beyond seed selection itself to determine optimal strategies for exploiting the identified seeds so as to sustain the spreading and prevent its decay [33], [34]. Furthermore, methods based on temporal point processes, e.g., to steer social activity both at a user (local) and network (global) level to maximise spread of the published content have also been proposed [35]. In turn, for influence suppression, node immunisation strategies represent a key line of research. These aim either to identify entities whose isolation would most effectively slow down the propagation [36], or to introduce a counteracting process in optimal parts of the network to stop the spread of a harmful one [37].

1.2 Key concepts

To ensure conceptual clarity of this thesis and eliminate potential ambiguities, three key concepts from the domain of spreading phenomena in multirelational systems are formally introduced below. The conclusions drawn from the following issues are gathered in Sec. 1.3 and then scattered into research gaps.

1.2.1 Multilayer networks

The most fundamental formalism underlying the research presented here is the multilayer network. Every multirelational complex system considered in this thesis is modelled using this structure (although it represents only one of many possible ways of expressing such systems, as mentioned in Sec. 1.1). Following [8], Def. 1.2.2 introduces this concept. Nevertheless, in order to highlight the advantages offered by multilayer networks, a classically formulated network (also referred to as the monoplex or the simple network) is described in Def. 1.2.1.

Definition 1.2.1 (Network). *A network can be described as a tuple $N = (V, E)$ comprising the following sets:*

- nodes $V = \{v_1, \dots, v_{|V|}\}$,
- edges $E = \{(v_i, v_j) : E \subseteq V \times V \wedge (v_i, v_j) \equiv (v_j, v_i)\}$.

Definition 1.2.2 (Multilayer network). *A multilayer network can be described as a tuple $M = (A, L, V, E)$ comprising the following sets:*

- actors $A = \{a_1, \dots, a_{|A|}\}$,
- layers $L = \{l^1, \dots, l^{|L|}\}$,
- nodes $V = \{\dots, v_i^m, v_i^n, v_j^m, v_j^n, \dots\} : V \subseteq A \times L$,
- edges $E = \{(v_i^m, v_j^m) : E \subseteq V \times V \wedge (v_i^m, v_j^m) \equiv (v_j^m, v_i^m) \wedge (v_i^m, v_j^n) \notin E\}$.

When analysing Def. 1.2.2, one can note that a multilayer network is a collection of networks defined according to Def. 1.2.1, where each actor may be represented by at most one node per layer, i.e. $V \subseteq A \times L$. For instance, the representation of actor a_i in layer l_m is given by the node v_i^m . Moreover, for the multilayer network comprising only one relationship $|V| \equiv |A|$ holds and, contextually, these sets are the same. The model disallows interlayer edges; that is, links can be established only between nodes within the same layer: $(v_i^m, v_j^n) \notin E$ for $m \neq n$. Furthermore, to simplify the analysis, all edges are assumed to be undirected and unweighted, meaning that $(v_i^m, v_j^m) \equiv (v_j^m, v_i^m)$. A toy example of a multilayer network is presented in Fig. 1.1. Note that not every actor is represented across all layers (e.g. a_{11} has its avatars on l^1 and l^2 , but there is no node v_{11}^3).

The additional modelling space provided by multilayer networks becomes evident when comparing Def. 1.2.2 with Def. 1.2.1. It can be seen that multilayer networks extend the modelling space by the additional dimension, allowing relationships to be distinguished explicitly. For instance, in a model of a social system, l_1 may represent physical encounters, while l_2 may capture interactions through online communication platforms. Moreover, a real-world multilayer network is not merely a coupled collection of monoplex graphs. Such structures exhibit complex intrinsic dependencies between their layers [38].

At this point, it is worth noting that the concept of multilayer networks partially overlaps with that of heterogeneous graphs [5], the latter offering a richer representation of the actor set (e.g. distinguishing between different node classes) and allowing interlayer connections

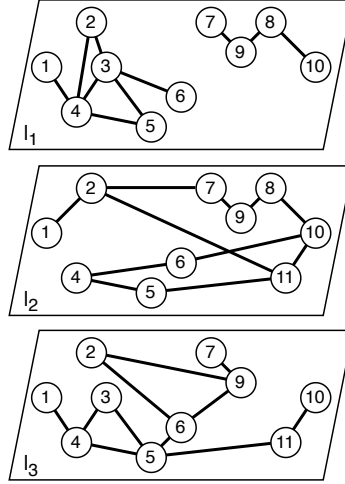


Figure 1.1: A toy example of a multilayer network with three layers, eleven actors, thirty nodes, and thirty two edges.

Another structurally similar model that is widely studied in the literature is that of multiplex networks [39], [40], which differ from Def. 1.2.2 by an additional constraint enforcing actors to be represented on every layer. An important remark here is that in cases when multiplex networks are more convenient structure, a common practice is to upsample the multilayer network with artificial isolated nodes for actors that are not represented in each layer.

1.2.2 Spreading phenomena in multilayer networks

Having formally defined how multilayer networks are understood in this dissertation, the next step is to elaborate on the dynamics that occur within such structures. In general, this dissertation adopts the spreading process as defined in Def. 1.2.3, which is sufficiently general to encompass virtually any diffusion model. It should be noted that the following formalism deliberately does not introduce an explicit agent-level-state-update function, as its form may vary depending on, e.g., whether nodes or actors in case of multilayer networks are the subjects of the spreading process.

Definition 1.2.3 (Spreading Model). *Let $\chi(M_{t-1}, T, W) \rightarrow M_t$ denote a spreading model, that is, a function defined on a network-like structure M , a finite set of possible states T that each agent may adopt, and a set of parameters W . The parameters specify the conditions under which state transitions occur as a result of interactions between neighbouring agents. An execution of the spreading model transforms the system from state M_{t-1} to state M_t .*

In principle, two essential components are required to describe diffusion dynamics in networks: the spreading model and the structure over which the propagation occurs. As observed in the survey on this topic [41], four general configurations of the spreading model and the underlying network can be distinguished for the area under consideration. In Fig. 1.2, two cases considered in this dissertation are highlighted: a single-spread process in a multilayer network and a multi-spread process in a multilayer networks.

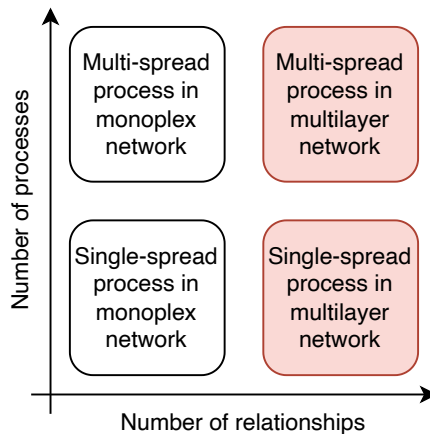


Figure 1.2: Possible alignments of spreading phenomena in multilayer networks [41]. Highlighted in red are cases considered in this work.

The former case, despite its simpler alignment, possesses its own sources of complexity, primarily related to structural constraints. Each of the sets comprising a multilayer network (Def. 1.2.2) introduces a separate degree of freedom. Most importantly, since actors constitute an abstraction over nodes, spreading phenomena can be modelled either at the actor level [42] or at the node level [43]. This lack of a unified approach makes it difficult to compare methods defined over spreading dynamics in multilayer networks. Nonetheless, as will be seen later, this remains a persistent challenge within the domain. The remaining components of multilayer networks — edges and layers, also play a similar role in spreading phenomena, as both can be weighted. For instance, in [44], the authors assign different weights to relationships (layers), within a node-based spreading model. The same naturally applies to edges, whose weights may reinforce or weaken transmission rates between neighbouring agents.

As for the second scenario, involving multiple diffusion phenomena propagating within a multilayer network, Bródka et al. [41] proposed a taxonomical framework that distinguishes four key components: (1) what spreads, (2) where the spreading takes place, (3) how the spreading process is modelled, including its interactions with other concurrent processes, and (4) why the propagation unfolds in a particular way. Each of these aspects is briefly summarised below to provide an overview of the problem landscape.

What spreads? refers to the phenomenon that propagates across the system, such as an opinion, a contagious microorganism, a behaviour, or a meme. This question sheds light on the specific model employed, as they are inherently domain-dependent. For instance, the Linear Threshold Model [45] is typically applied to simulate the dissemination of opinions, whereas the Susceptible-Infected-Recovered [46] is used in epidemiological contexts. In this regard, the literature appears relatively homogeneous: according to [41], approximately 95% of studies investigating multispread in multilayer networks consider either the spread of multiple diseases or the interplay between disease and awareness.

The second component — Where does it spread? — concerns the medium through which

the propagation occurs. Six factors can be distinguished in this context: the network topology, the number of layers, the multiplexity of the network, the allowance of interlayer connections, and the type of external influence (i.e., whether the system is isolated or not). A key observation is that the vast majority of studies focus on two-layer networks, with only around 5% examining networks with more than two relation types [41].

How does it spread? examines the characteristics of spreading models and the ways in which they interact with one another. This component addresses questions such as whether the process can switch layers during propagation, whether agents can be co-infected (i.e., simultaneously acquire multiple pieces of information), whether phenomena spread synchronously, and what forms of interaction may occur between processes. Regarding the latter, four scenarios are distinguished: supporting processes, competing processes, and mixed approaches (both competitive and collaborative interactions simultaneously), or no interaction at all. The second and third scenarios account for more than 80% of the literature [41].

The last question: Why does it spread in that way? addresses the underlying factors and mechanisms that determine the observed propagation patterns. It provides an analytical perspective on the assumed spreading scenario, aiming to explain why diffusion occurs as it does. For example, one may investigate the influence of multilayer network properties on the spreading process, such as how correlations between node degrees affect overall diffusion effectiveness, or examine the impact of process-specific characteristics, for instance how interactions between competing or cooperating processes modify epidemic thresholds.

1.2.3 Influence maximisation

The influence maximisation problem was originally formulated by Kempe et al. [28] more than twenty years ago and has since remained a subject of interest amongst researchers. In its classic form, it operates on standard networks composed of links between pairs of nodes (Def. 1.2.1), each of which can adopt one of two states: inactive or active, according to a specified spreading model (Def. 1.2.4). The spreading model reproduces a particular propagation mechanism of a given phenomenon (e.g., gossip) and is driven by interactions between neighbouring nodes. Finally, the influence maximisation problem concerns identifying a set of nodes which, when used to initiate the spreading process, yields maximal propagation effectiveness (Def. 1.2.6).

Definition 1.2.4 (Binary System). *Let a be an agent represented in the network-like structure. At time-step t , the agent can adopt one of two discrete states, $y_a(t) \in \{0, 1\}$: inactive ($y = 0$), and active ($y = 1$) as a result of the spreading model.*

Definition 1.2.5 (Active Sets). *Let S_t be a set of active agents in the binary system at the time step t . We denote S_0 as a seed set, i.e. a set of initially active agents before any dynamics occur in the system. When $S_{t-1} = S_t$ the system reaches a steady state.*

Definition 1.2.6 (Influence Maximisation Problem). *Let the influence of an agent set S be a*

function $\sigma(S) \rightarrow \mathbb{N}^+$ which denotes the expected number of active agents in a binary system at the steady state, given that S is used as a seed set. The influence maximisation problem asks, for a parameter s , to find an agent set of the maximum influence, i.e., $S : \arg \max(\sigma) = S \wedge |S| \leq s$.

These definitions nonetheless require additional commentary. Since the influence maximisation problem was originally formulated for monoplex networks, the theorems presented above have been adjusted to abstract from that specific structure (e.g., “nodes” are replaced with “agents”), while remaining consistent with the original formulation.

The influence maximisation problem is known to be NP-hard [28]. Consequently, the majority of studies have addressed it experimentally by proposing various methods for selecting a sub-optimal seed set. In recent years, however, numerous alternative approaches have emerged, often accompanied by differing interpretations of what influence maximisation entails. Although some researchers have attempted to categorise these approaches [47], no universally accepted taxonomy has yet been established. Fig. 1.3 presents a diagram outlining the potential components of the problem, each of which can be considered independently and affects the overall diffusion effectiveness.

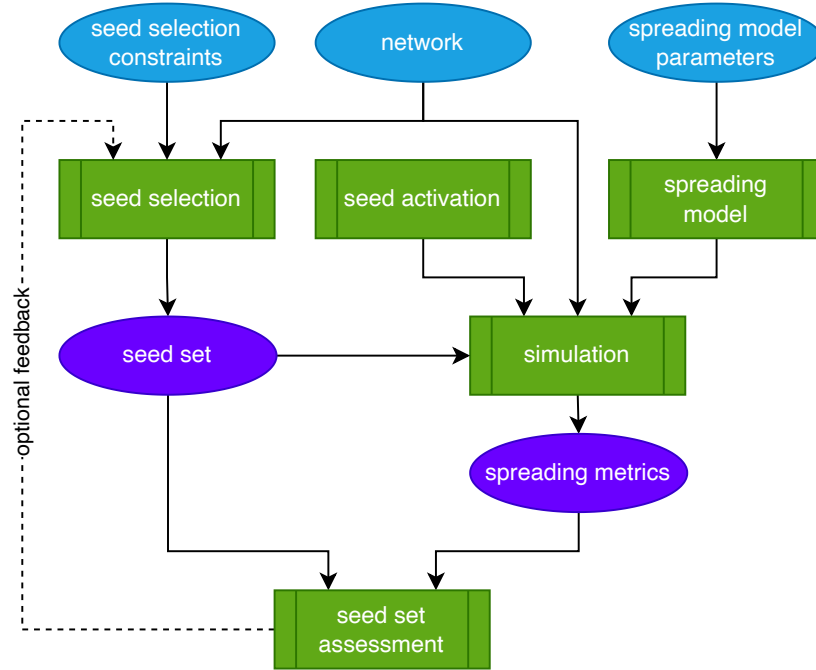


Figure 1.3: Components of the influence maximisation problem in its novel form depicted as a diagram. Rectangles denote elements sensitive to the considered case mechanisms, pale blue ellipses input data, and dark blue ellipses intermediate data structures [P-4].

As one can observe, the influence maximisation problem has evolved considerably over the years through the introduction of new components to its original formulation. Its present form depends on several factors that shape the spreading dynamics within a network and, consequently, determine the overall effectiveness of the diffusion. Naturally, the network itself plays a crucial role: it may be directed, static, temporal, multilayer, or hyper-relational. Moreover, within each of these frameworks, various geometrical features (such as the degree

distribution) can also have significant influence. Another key factor is the choice of the spreading model (e.g., Independent Cascade, q -Voter, or Susceptible-Infected-Recovered [46]) and its parameters (e.g., the infection rate β in the latter). Since the spreading model defines the propagation mechanisms, it constitutes a central element of the problem. Additionally, the method used for seed selection (e.g., random choice, greedy strategy, or degree-based ranking), the manner in which seeds are activated (simultaneously or sequentially during propagation), the constraints imposed on seed selection (e.g., budget, cost, or time), and the metric of influence to be optimised (for instance, the final number of activated agents or the number of simulation steps required to activate 50% of them) are all important considerations that cannot be neglected. Moreover, these factors are often interdependent; for example, the seed selection method may depend on the adopted spreading model. The number of degrees of freedom in the problem is therefore substantial, which explains why numerous studies have been devoted to it and why researchers still consider this problem as interesting.

1.3 Challenges within the field

Following the main concepts on which this work is grounded, the primary challenges that must undertake when studying spreading phenomena in multilayer networks are introduced. They constitute the foundation upon which the specific research gaps are subsequently formulated in Sec. 1.4. These are:

- modelling of spreading phenomena in multilayer networks,
- methods for maximising influence in multilayer networks,
- tools and datasets complementing the above aspects.

The first challenging area is fundamental, as it relates to the ability to accurately simulate diffusion processes. As noted in Sec. 1.2, the mechanisms underlying various spreading phenomena differ (e.g., the propagation of a contagious disease versus the dissemination of political opinions). Moreover, real-world systems are often subject to multiple, simultaneously unfolding processes. When multirelational structures are considered, these mechanisms require further adaptation, as the networks themselves are more complex and incorporate additional structural and functional dimensions. Although the structural factors that influence single-spread can be named, the field still lacks a unified and widely accepted framework for modelling diffusion phenomena in multilayer networks. This absence of methodological coherence complicates efforts to conduct comparative studies and, more importantly, hinders the development of techniques that rely on such dynamics, amongst which influence maximisation is a prominent example. Furthermore, while a coarse taxonomy of multi-spread processes already exists, the issue of modelling dependencies between coexisting processes (e.g., competitive, cooperative, or neutral) remains unresolved. Finally, another challenge lies in achieving an appropriate balance between the generality and specificity of spreading models: the former allows results to remain applicable across diverse contexts, whereas the latter is necessary to capture the

particular characteristics of a given phenomenon with sufficient realism.

As for the second area, analysing all factors that affect the problem of influence maximisation (Fig. 1.3) would involve an amount of work far exceeding the scope of a single doctoral dissertation. Moreover, addressing these factors sequentially does not guarantee solving the problem completely, as methods tailored to a specific configuration of components are usually not transferable to others. For instance, Erlandsson et al. [48] and Zhao et al. [49] demonstrated that applying influence maximisation techniques developed for monoplex networks does not ensure success in multilayer setup. Similarly, Kitsak et al. [50] showed that while the k-shell index effectively identifies influential single spreaders, the efficiency of multiple spreaders depends critically on their mutual distance, as closely located seeds may produce overlapping spreading regions and thus reduce the overall coverage. These examples highlight the complexity of the problem and underscore the challenging task of finding a representative subset of factors to analyse within a single thesis. Another persistent yet interesting challenge would be to utilise the power of computer science and attack the problem from various perspectives. This direction may prove particularly fruitful when focusing on aspects such as developing new methods for seed selection, or designing frameworks within which network-like systems can be controlled and evaluated in terms of spreading effectiveness. The former may involve a wide range of techniques, from simple heuristics to advanced deep learning-based strategies. The latter encompasses approaches for benchmarking seed selection methods and assessing how and why a given network-like system is prone to the spread of a particular phenomenon; in other words, establishing objective metrics and comparison methodologies.

Finally, the third aspect (i.e., complementary tools and datasets), although not strictly scientific and therefore auxiliary, deserves appropriate emphasis. It is natural that any computational research must be conducted using suitable environments and based on appropriate data. However, a problem arises when available toolkits are not standardised or do not exist at all, and the lack of source code and data accompanying published articles becomes, paradoxically, an implicit standard. This is, partly due to its interdisciplinary nature, the case in network science. A substantial volume of research in which source code and datasets remain inaccessible significantly undermines the credibility not only of individual studies but of the discipline as a whole. Therefore, the last identified challenge is to implement the postulates advocating greater openness and reproducibility into the studies comprising this dissertation. In other words, to conduct research with particular attention to the reusability and accessibility of the resulting artefacts, such as datasets, experimental environments, and implementations of proposed methods.

1.4 Identified research gaps

Despite the fact that the problem of spreading phenomena and influence maximisation in multilayer networks is already relatively well developed, several issues have not yet been

sufficiently explored. Following the state-of-the-art review, seven research gaps have been identified with the aim of addressing them in the course of this doctoral study. They are outlined below in brief and elaborated upon in the subsequent sections.

Research Gaps

- RG-A** Spreading in multilayer networks has not been systematised from a modelling perspective, nor has influence maximisation in such structures been approached from an actor-based viewpoint.
- RG-B** Metrics used to assess spreading effectiveness do not consider the conditions under which diffusion is triggered, nor the holistic properties of the spreading regime.
- RG-C** In studies of spreading phenomena in multilayer networks, contact-based datasets are under-represented, while synthetic graph generators exhibit notable limitations.
- RG-D** There is no comprehensive perspective on the overall effectiveness of rank-refining influence maximisation methods in multilayer networks.
- RG-E** Controllability-based seed selection methods have not yet been evaluated in the multilayer setting.
- RG-F** Machine learning-based influence maximisation approaches do not fully exploit the generalisation capabilities they potentially offer.
- RG-G** Techniques for maximising influence in multilayer networks do not leverage possible interventions on the system’s structure.

1.4.1 Research gap A

Statement: *Spreading in multilayer networks has not been systematised from a modelling perspective, nor has influence maximisation in such structures been approached from an actor-based viewpoint.*

Discussion: The complexity of spreading phenomena in multilayer networks involves an additional degree of freedom compared with those occurring in monoplex graphs. As stated in Sec. 1.2.2, a guideline for categorising them has been recently proposed. Nonetheless, the state-of-the-art works do not address the problem of developing a coherent, universal approach to modelling these processes. Following [41], studies on spreading in multilayer networks typically employ custom approaches tailored to the specific problem addressed in each work.

The same applies to works on influence maximisation in those structures: it is usually framed within the node-based spread, without revisiting the assumptions underlying the applied propagation model [43], [49], [51], [52]. Following Sec. 1.2.1, the biggest advantage brought by the emergence of multilayer networks is their ability to aggregate the same

entities across various relationships. Therefore, validating the methodological correctness of considering the problem either from the perspective of nodes or from that of actors before proposing new methods addressing this problem would be beneficial.

Next, the issue touched upon earlier (Sec. 1.3), namely the lack of an environment in which simulations can be conducted, is also important, albeit more technical. Since multilayer networks are not a typical data modality used in industry, there is no toolkit that can be employed in works addressing the topic considered here. Three frameworks most closely matching the area of concern are: **NetworkX** [53], **NDLIB** [54] and **multinet** [55]. The first allows for network manipulation in Python and is deeply integrated into the data science technological stack. Nonetheless, **NetworkX** provides neither mechanisms for simulating spreading phenomena nor functionality for manipulating multilayer networks. **NDLIB** addresses the former issue, as it is designed for modelling diffusion processes over graphs, but it does not support multirelational ones. The third one — **multinet**, is a versatile toolkit for processing multilayer networks, but it does not provide any functionality for studying their dynamics.

Finally, this gap has not been formulated as the first one by chance. It underlies every subsequent gap (with the exception of **RG-C**). Regarding the requirements for addressing it, there are two main aspects. Foremost, it ought to provide a model that allows the description of universal patterns of interactions between spreading phenomena (supporting, competing, indifferent, etc.). The second is an evaluation of how the influence maximisation problem in multilayer networks should be understood from the actor level. At this point, it would be preferable to identify and adapt an existing mechanism that can subsequently be reused in studying various spreading models. Finally, as this gap affects the entire venture, it is expected to be revisited when addressing the consecutive research questions.

1.4.2 Research gap B

Statement: *Metrics used to assess spreading effectiveness do consider the conditions under which diffusion is triggered, nor the holistic properties of the spreading regime.*

Discussion: As noted in Sec. 1.1, the effectiveness of influence maximisation methods is typically measured by the number of activated nodes produced by a given seed set. This stems from the influence function σ (Def. 1.2.6). One of the advantages it offers is the property of submodularity, which was utilised by Kempe et al. [28] to provide the approximation guarantee of the optimal solution for the greedy hill-climbing routine proposed therein.

Despite the substantial benefits of formulating the influence maximisation task within the framework of optimising the influence function, it is important to recall that this formulation was originally proposed for spreading models within the monoplex setting. Moreover, in experimental studies that employ methodologies too complex to permit a theoretical assessment of their performance, the property of submodularity carries less practical value. Nevertheless,

works addressing influence maximisation in multilayer networks typically rely on this metric to evaluate the effectiveness of their approaches, even though these studies are predominantly based on simulation methodologies [43], [49], [51], [52]. However, measuring only the number of activations without considering the seed set size obscures comparisons between different methods across and spreading regimes. It is not difficult to see that activating 60% of the network with a seed set of size 5% constitutes a greater achievement than doing that by initiating diffusion from 10% of the agents.

Recalling again Sec. 1.1, maximising spread can be interpreted as a form of control over the phenomenon. Thus, instead of assessing only the ultimate penetration of the process obtained from a given seed set, one may also account for its intermediate dynamics, which are, in some contexts, of primary importance. In combating an epidemic (such as COVID-19), it can be more critical to flatten the infection curve in order to preserve the capacity of the healthcare system than to minimise the total number of infections, as most of the population will sooner or later contract the disease [56]. Analogously, in politics, the objective is to persuade the largest possible fraction of citizens before the election occurs. Hence, a viral campaign must reach the maximal share of voters not by the time it decays, but by the election day. Studies considering dynamical properties of influence maximisation techniques are less common, but not non-existent. For instance, [57] compares seed selection strategies by counting the number of simulation steps required for the process to cover the network, although this analysis is conducted in a monoplex setting. For multilayer networks, an interesting concept is introduced in [48], where the area under the activation curve is used. Its advantage lies in reducing the entire curve to a single scalar, which is easier to handle. Nonetheless, the metric is defined for node-based spreading and would therefore require adaptation to an actor-centric diffusion.

In summary, this research gap reveals a need for a comprehensive set of metrics by which influence maximisation techniques in multilayer networks can be assessed. These metrics should consider both the conditions under which experiments are performed (e.g., the seed set size) and should take into account both the total penetration of the process and its dynamics.

1.4.3 Research gap C

Statement: *In studies of spreading phenomena in multilayer networks, contact-based datasets are under-represented, while synthetic graph generators exhibit notable limitations.*

Discussion: Spreading phenomena occur in a wide range of systems (Sec. 1.1). Therefore, employing appropriate datasets in experimental analyses is essential for drawing meaningful conclusions. With regard to this problem, two principal types of data can be distinguished: synthetic and real-world.

The latter is usually considered in the social context (but not exclusively), and networks representing human interactions predominate. Nonetheless, these networks are usually scraped

from the Internet or at least from computer-based systems due to the ease of constructing the underlying graphs and the availability of raw data. There are many examples of such datasets, for instance this website: <https://manliodedomenico.com/data.php>, which, as of 13th November 2025, contains about 50 different multilayer networks, the majority of which are based on online interactions. However, datasets built upon real, physical encounters are harder to obtain and are therefore relatively rare, especially the multilayer ones. The most well-known is the 70-year-old *ckmp* [58], which was built upon physicians in Illinois and the impact of social ties on their adoption of a new medicine. Another, also widely used, network is *aucs* [59], which presents both social and online interactions between employees of a university but is more than 10 years old. One can also mention the *lazega* network [60] (published almost 20 years ago), which is also based on physical interactions. Therefore, the need for new networks built on physical human contacts is evident. Utilising them can furthermore balance the Internet-based social systems, which are much better represented amongst the datasets.

As for the synthetic data, studies on spreading phenomena utilise them either by coupling independently generated graphs or by employing synthetic models that account for interlayer dependencies. The former approach, although much more limited, is still a very popular technique [51], [61], [62]. The latter usually aims to address the issue either by modelling community alignment (i.e., groups of nodes that are densely connected amongst themselves while being sparsely connected to nodes outside [63]) or by considering structural correlations across the layers (e.g., degree distribution). Over the years, many approaches have been proposed to generate synthetic networks with interlayer dependencies [64], [65], [66]. These works naturally influence studies on spreading phenomena, as using such models alongside real-world networks allows researchers to generalise conclusions and helps in cases of data scarcity. Notably, one of the generators that can be easily applied to the problem of influence maximisation is **multinet** [55]. It enables the modelling of networks whose layers follow either preferential-attachment or Erdős-Rényi models and exhibit interlayer dependencies, although this is modelled implicitly. It has even been successfully used to tackle the problem of influence maximisation [67]. Nevertheless, **multinet** model can be insufficient if one wishes to examine the control of spreading through structural manipulation of the network (see Sec. 1.4.7), as it offers only basic representations of interlayer dependencies. Other models may also be insufficient, as they present their own limitations. The approach proposed by [64] models exactly matching groups across the layers; [65] models only the coupling of nodes and partitions; and [66], despite being the most advanced, exhibits computational drawbacks.

In summary, the discussed research gap reveals the following shortcomings. Foremost, it is necessary to consider data representing various systems. As for the social ones, networks constructed from physical encounters should be taken into account and if possible acquired, as they provide a counterbalance to common Internet-based networks. Next, it is important to include synthetic graphs to provide a wider range of results and enable comparisons, but these graphs should capture interlayer dependencies. Finally, for the problem of controlling

spread through the topological properties of the network, there is a need for a more advanced and efficient synthetic models.

1.4.4 Research gap D

Statement: *There is no comprehensive perspective on the overall effectiveness of rank-refining influence maximisation methods in multilayer networks.*

Discussion: As illustrated in Fig. 1.3, there are numerous ways to maximise influence. Nonetheless, the canonical perspective on this problem is to identify an appropriate seed set that is used to trigger diffusion (see Def. 1.2.6). The most fundamental approaches to this issue are those relying solely on the structural information of the network. According to the recent survey paper on influence maximisation techniques, such methods are grouped under the category of rank-refining heuristics [47]. They aim to assign a score value for each agent (according to given heuristic) so that a k -element seed set can be obtained by selecting the k -long head of a ranking created with respect to those scores.

The identified research gap concerns the absence of a comprehensive effectiveness evaluation of such methods with respect to both network structure and spreading parameters. Many studies proposing new methods for multilayer networks (e.g., [48], [68], [69], [70]) do not account simultaneously for a broad spectrum of these factors, primarily due to the computational effort required to conduct such an extensive analysis. Nonetheless, one could hypothesise that the effectiveness of methods belonging to the same class may be approximately similar, or at least that they should exhibit consistent trends with respect to experimental conditions (such as increasing performance with growth of link-based activation probability in the Independent Cascade Model). Therefore, it can be argued that the their performance cannot be meaningfully discussed without considering the broad experimental conditions; if performance differences occur, they should be interpreted globally. For example, one might observe that “method A” performs better up to a certain threshold value in the Linear Threshold Model, while “method Y” surpasses it beyond this point. Following such an approach would allow for a systematic comparison of their relative performance.

Another aspect concerning this class of methods is their connection to the problem of measuring centrality in networks. It aims to quantify the importance of agents with respect to their position within the network: those that are more central or structurally crucial are assigned higher values of the given metric [71]. The research gap discussed here relates closely to the notion of centrality in multilayer networks as filling it in can contribute to this problem as well. Notably, due to the relative novelty of these structures, there is still no consensus on how to extend many centrality algorithms. For instance, one of the most fundamental centralities — degree, can be interpreted in the multilayer setting in three ways: (1) the very same as in monoplex networks, by treating nodes on separate layers as entirely distinct entities; (2) by counting all connections of nodes representing a given actor across layers; or

(3) by counting the number of actors linked to a given actor (the neighbourhood size).

In summary, the discussed research gap can be addressed through a broad evaluation of rank-refining seed selection methods for multilayer networks conducted under equally broad experimental conditions. Such an evaluation would allow for the identification of baseline methods against which more sophisticated approaches can be compared in future studies.

1.4.5 Research gap E

Statement: *Controllability-based seed selection methods have not yet been evaluated in the multilayer setting.*

Discussion: As stated in Sec. 1.1, maximising influence can be seen as a form of controlling spread, which bridges it into the distinct realm of system controllability. That field is only partially related to networks: its typical problems cover such cases as steering physical systems towards a desired state. A straightforward example can be drawn from thermodynamics, such as maintaining a comfortable temperature in a household. Two centuries ago, it was still common practice for every room to have its own source of energy (e.g., a tiled stove), as it was easier to build such an installation. Despite the fact that central heating was widely used by the Romans [72], only recently has this form of facility become common. Considering that invention from a control point of view, it is equivalent to introducing a central entity in the system which, through links to the peripheries (either by water pipes, as now, or by hot-air channels in antiquity), allows exerting governance over the entire system.

Networks have also been the subject of studies within this domain [73]. One of the most prominent problems in this area is identifying a minimal set of so-called driver nodes, which are capable of steering the network towards a desired state [30]. This problem has been examined, for instance, in the context of structural controllability in large-scale dynamical systems [74]. Following the idea that maximising spread can be interpreted as a form of system control, it is natural to consider driver nodes as effective diffusion seeds. Studies verifying this hypothesis in monoplex networks have yielded promising results [57]; however, no attempts have been made so far to extend this line of research to multilayer networks.

This observation highlights and confirms the existence of the research gap. Nonetheless, two key challenges arise when attempting to address it. First, as spreading processes in multilayer networks can be examined from either the node- or actor-level perspective, it is necessary to align the notion of driver “agents” with the diffusion model employed. The concept of the Minimum Dominating Set (i.e., a set of minimum cardinality consisting of driver nodes) [75] has already been extended to the multilayer scenario, though only from a node-based perspective [76]. Consequently, for actor-based influence maximisation, it would first be necessary to redefine this concept. The second challenge concerns how to incorporate driver nodes or actors into the seed selection process, as its effectiveness is typically evaluated

under a given budget (see Def. 1.2.6). Therefore, an appropriate methodology should be developed to assess the general applicability of network-control mechanisms to influence maximisation, rather than relying on particular heuristics built upon them.

1.4.6 Research gap F

Statement: *Machine learning-based influence maximisation approaches do not fully exploit the generalisation capabilities they potentially offer.*

Discussion: Traditional approaches to influence maximisation have predominantly relied on heuristics (Sec. 1.2.3). Nonetheless, the recent flourishing of machine learning approaches has also affected the domain of network science, resulting in the emergence of works addressing this problem. One of the pioneering methods among them is [52], in which the authors propose a framework named PIANO (a.k.a. DISCO) that utilises latent node-level embeddings and reinforcement learning to find a sub-optimal seed set. Although several years have passed since the initial version of this method was published, these trends remain popular, and studies aiming to push forward either the effectiveness or efficiency boundaries continue to appear in the academic literature [77], [78]. However, their major shortcoming is the inability to generalise to networks unseen during training.

As for multilayer networks, or even within a broader setup of heterogeneous graphs, the number of state-of-the-art machine learning-based methods is incomparably smaller. One exemplary approach is GBIM [79], which frames multiplex influence maximisation as a graph Bayesian optimisation problem, but considers it only in a narrowly defined two-layer network with a specific spreading model. At this point, it is also worth mentioning certain techniques for ranking prediction tasks in heterogeneous graphs [80], [81], as the problem of seed selection can easily be treated in a similar manner. Nonetheless, again the issue of being unable to process unseen data arises. The same applies to unsupervised actor-embedding methods that are potentially applicable in this context, e.g. [11].

In conclusion there are two main pleas shaping this research gap. First, machine learning-based influence maximisation has been explored only marginally in the context of multilayer networks. Second, the potential of inductive learning (enabling models to generalise across unseen network structures without retraining) has yet to be realised in this domain. Addressing these challenges could yield models capable of both outperforming existing methods and offering practical advantages through their reusability across diverse networked systems.

1.4.7 Research gap G

Statement: *Techniques for maximising influence in multilayer networks do not leverage possible interventions on the system's structure.*

Discussion: The final research gap concerns an alternative perspective on maximising influence. Following Fig. 1.3, the problem comprises not a single element but a collection of interdependent components. However, in its canonical formulation, it is the composition of the seed set that ultimately constitutes the solution. While this brings several advantages (see Sec. 1.4.2), it narrows down the range of possible approaches towards the problem.

Graph theory has coined the concept of expanding graphs (not the same as expanders, although the latter may also be expanding under certain conditions) [82]. These structures are characterised, i.a., by the property that connected sets possess a neighbourhood which is coarsely in the order of the average degree larger than the set itself. This makes them a conducive environment for information spreading solely on the basis of their topology. To the best of the author’s knowledge, this concept has not yet been examined in the context of multilayer networks, even though it delineates a premise that can be exploited to achieve more effective spread.

Concurrently, another concept relevant to understanding this gap has emerged — digital twins. These are synthetic models of physical systems, employed for monitoring, maintenance, and analysis. A key characteristic is the bidirectional transfer of information between the real object and its digital counterpart [83]. In other words, a digital twin comprises three main components: the physical structure, the virtual model, and the connection between them [84].

Civil engineering is one of the domains in which this concept has been successful [83]. For instance, a recent study demonstrates its applicability in managing a railway bridge maintenance [85]. Robotics is another field where its utility has been proved. One example, in which the author was involved, concerns the development of deep learning models for controlling an autonomous underwater vehicle. These models were trained in a synthetic environment a close twin of the reservoir where the vehicle was intended to operate [86]. An iterative exchange between performance “in the wild” and in the “virtual world” enabled improvements in the vehicle’s reliability. More recently, an attempt has been made to categorise digital twins of networked systems, with the authors distinguishing four levels of modelling complexity for such objects [87].

Therefore, on the basis of these two concepts, it is possible to outline this research gap. Namely, rather than focusing on identifying the most spreading-prone agents within a network, it may be more effective to consider modifying its topology to facilitate spreading. Such modifications, however, should be performed at a global level, as local interventions (e.g., edge rewirings between communities) can be too tedious and case-specific to yield generalisable conclusions. This macro level modelling entails the use of a synthetic graph generator, with the original network treated as the “real object” and, as its “digital twin”, the set of configuration parameters proper to the generator enabling the creation of a similar network. With this setup, it would be possible to conduct global interventions that reinforce the spreading process.

At this point, it is clear that addressing this gap requires an expressive synthetic graph generator. It should allow the modelling of interlayer dependencies inherent to multirelational

systems and provide a rich parameter space in which the network under evaluation can be embedded. However, as noted in Sec. 1.4.3, no such framework is currently available. Furthermore, a methodology for determining such a configuration needs to be developed as well, as it closely depends on the generator employed. Therefore, due to the broad spectrum of sub-issues that together constitute a separate research problem of big scale, a feasibility study for this approach should be undertaken as part of this work.

1.5 Innovation of the research

Having defined the research gaps in the field of spreading phenomena in multilayer networks, it could be beneficial to elaborate on the innovative aspects of the research aimed at addressing them. These primarily lie in the fact that, by tackling these problems, deeper insights into complex, multirelational systems may be gained. Moreover, utilising the lens of computer science provides additional understanding of the mechanisms underlying dynamical processes in these networks, by its complementary perspective.

Since multilayer networks are a relatively recent concept, examining phenomena that spread within them resonates across many aspects of modern life. For instance, the influence maximisation problem can be directly translated into both business and social contexts, depending on the nature of the system represented by the analysed network. In the case of data originating from social media platforms, computational tools and methods can be developed or adapted to create systems for identifying so-called influencers. Engaging such individuals in marketing campaigns, coordinated simultaneously across multiple platforms, may substantially enhance consumer awareness of advertised products and increase purchase rates. In the case of epidemiological data, such frameworks make it possible to forecast the spread of diseases and identify the most vulnerable groups. Moreover, when the network under consideration spans multiple types of relationships, researchers can evaluate potential policies for suppressing contagion and provide decision-makers with better strategies. In other words, by employing multilayer networks, situations similar to the COVID-19 pandemic, during which many countries imposed lockdowns, could potentially be managed in a milder form, by taking into account the subtle nuances of the different social bonds people maintain. The value of this line of research extends further through its applicability to more complex scenarios. This is particularly evident when addressing co-occurring phenomena, like the simultaneous spread of a disease and of opinions about it, which directly influence individual decisions, such as the willingness (or reluctance) to get vaccinated or to limit social interactions.

One may also consider the innovative aspect this dissertation from the opposite perspective: namely, how studying networks contributes to the development of computer science itself. The answer can be found by comparing networks with other data modalities. Images, text, audio, or electrical signals are typically represented by data structures characterised by a high degree of regularity; e.g., a raster image corresponds to a uniform grid. In contrast, the vast

majority of networks are inherently irregular. Hence, efforts to understand the phenomena emerging in such structures (rather than exploiting the regularities of more conventional data) facilitate the generalisation of many task-specific methods. Convolutional neural networks [88] serve as a prominent example: originally developed for image processing, they have since been successfully extended to irregular structures in the form of graph convolutional networks [89], thereby opening a new chapter in the development of machine learning.

Finally, the identified research gaps highlight intersections between various paradigms and the network-based problems addressed in this dissertation. This was intentional and also embodies an innovative aspect. When the research leading to this dissertation began, this very idea served as the guiding principle: to bring network science closer to Information and Communication Technology, while simultaneously translating well-established computational methods into the network domain.

Chapter 2

Research Goal

In this chapter, the primary focus is on stating the research goal along with resulting prerequisites and objectives. Next, the list of scientific papers that constitute this dissertation is provided. Finally, for each research item, a summary is given of how it was addressed by the respective articles.

2.1 The goal, resulting objectives, and prerequisites

As indicated above, this dissertation focuses on the problem of maximising influence in multilayer networks. To narrow the scope of the research comprising the thesis, it is necessary to formulate the overarching goal that delineates the path along which the following studies seek to provide an answer:

Develop and evaluate approaches for maximising influence in multilayer networks by leveraging selected paradigms that provide complementary perspectives on complex systems: a notion of actor centrality, controllability means, graph-based machine learning, and structural interventions on the analysed system.

In order to address the introduced goal, four subsidiary research objectives have been proposed (**RO-1–RO-4**). However, because the problem domain is broad and can be approached in multiple ways, it was necessary to accompany them with three research prerequisites (**RP-1–RP-3**). These prerequisites serve to ground the works addressing the objectives within a common conceptual field and to ensure the use of a unified set of methods, thereby making the research venture coherent.

Research Objectives

RO-1 Evaluate various rank refining methods for influence maximisation in multilayer networks and assess how they differ in terms of spreading effectiveness (from **RG-D**).

- RO-2** Propose a seed-selection approach that leverages the minimal dominating set for the influence maximisation problem in multilayer networks and verify its effectiveness (from **RG-E**).
- RO-3** Develop an inductive graph neural network that predicts which actors in a given multilayer network are the most effective spreading seeds, and validate its performance against other state-of-the-art methods (from **RG-F**).
- RO-4** Design a method to manipulate the structure of a multilayer network at a macro scale, enabling comparison of spreading effectiveness between the original and modified networks, and identify the structural properties that facilitate the influence maximisation process (from **RG-G**).

Research Prerequisites

- RP-1** Identify various methodologies for modelling spreading phenomena in multilayer networks, and select a single mechanism under which all research objectives will be evaluated (from **RG-A**).
- RP-2** Develop metrics for quantifying spreading effectiveness in multilayer networks, enabling comparison of spreads triggered under varying conditions (from **RG-B**).
- RP-3** Provide multirelational data that can serve as a structural backbone for experiments on influence maximisation and develop methods capable of real-time augmentation (from **RG-C**).

All of the above-mentioned research items (i.e., both prerequisites and objectives) result from an in-depth literature review, which enabled the identification of gaps in the current body of knowledge on influence maximisation in multilayer networks. The overall plan for addressing them involved conducting a series of studies that build upon one another. However, as the research prerequisites and objectives are interdependent, it was not possible to execute this plan in a strictly linear order. Accordingly, in line with the stated goal, the research objectives are treated as overarching components, while the research prerequisites are implemented at a pace determined by them.

2.2 List of publications

This thesis is a series of thematically related works, presented in the form of seven scientific publications. Their description is provided below, with bibliometric data according to the university's DONA database (<https://dona.pwr.edu.pl/szukaj>) and the corresponding CORE rankings as of the time of publication (<https://portal.core.edu.au/conf-ranks>):

- P-1** M. Czuba and P. Bródka, ‘*Simulating Spreading of Multiple Interacting Processes in Complex Networks*,’ in: 2022 IEEE 9th International Conference on Data Science and

Advanced Analytics (DSAA), pp. 1–10, Oct., 2022, DOI: 10.1109/DSAA54385.2022.10032425. **140 MNiSW points¹, Core A²**;

P-2 M. Czuba, M. Nurek, D. Serwata, Y.-X. Qi, M. Jia, K. Musial, R. Michalski and P. Bródka, ‘*Network Diffusion – Framework to Simulate Spreading Processes in Complex Networks*,’ in: *Big Data Mining And Analytics*, pp. 1–13, Aug., 2024, DOI: 10.26599/BDMA.2024.9020010. **20 MNiSW points, IF 6.2³**;

P-3 M. B. Paradowski, N. Whitby, M. Czuba and P. Bródka, ‘*Peer Interaction Dynamics and Second Language Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social Network Analysis*,’ in: *Language Learning*, vol. 74, no. S2, pp. 58–115, Dec., 2024, DOI: 10.1111/lang.12681. **200 MNiSW points, IF 4.2**;

P-4 M. Czuba and P. Bródka, ‘*Rank-refining seed selection methods for budget constrained influence maximisation in multilayer networks under linear threshold model*,’ in: *Social Network Analysis and Mining*, vol. 15, no. 1, pp. 46–25, Apr., 2025, DOI: 10.1007/s13278-025-01454-7. **100 MNiSW points, IF 2.8**;

P-5 M. Czuba, M. Jia, P. Bródka and K. Musial, ‘*Applicability of the minimal dominating set for influence maximization in multilayer networks*,’ in: *Journal of Complex Networks*, vol. 13, no. 6, cnaf036, Dec., 2025, DOI: 10.1093/comnet/cnaf036. **100 MNiSW points, IF 1.5**;

P-6 M. Czuba, M. Stolarski, A. Piróg, P. Bielak and P. Bródka, ‘*Identifying Super Spreaders in Multilayer Networks*,’ in: *Frontiers in Artificial Intelligence and Applications*, vol. 413 (ECAI 2025), pp. 2195–2202, 2025, DOI: 10.3233/FAIA251060. **140 MNiSW points, Core A**;

P-7 Ł. Kraiński, M. Czuba, P. Bródka, P. Prałat, B. Kamiński and F. Théberge, ‘*Multilayer artificial benchmark for community detection (mABCD)*,’ in: *Expert Systems with Applications*, vol. 307, p. 130 920, 2026, DOI: 10.1016/j.eswa.2025.130920. **200 MNiSW points, IF 7.5**.

2.3 Addressing the research items

Let us now discuss how each research objective and prerequisite are addressed by the articles constituting this dissertation. As already noted, the mapping between these items is not linear. Therefore, we first present a simplified diagram (Tab. 2.1) that illustrates these relationships in chronological order.

¹Points from the scored list of journals and conference proceedings published by the Ministry of Science and Higher Education of the Republic of Poland (ascending scale from 0 to 200).

²Category of a computer science and engineering conference by the Computing Research and Education Association of Australasia (descending scale from A* to C).

³Impact Factor, a journal ranking metric calculated by the Web of Science (ascending numerical scale).

Table 2.1: A matrix illustrating the relationships between research gaps, items and the articles addressing them, arranged in chronological order of execution.

	P-1	P-2	P-3	P-4	P-5	P-6	P-7
RG-A → RP-1	■	■		■			
RG-B → RP-2				■	■		
RG-C → RP-3			■	■			■
RG-D → RO-1				■			
RG-E → RO-2		■			■		
RG-F → RO-3						■	
RG-G → RO-4							■

Time flow →

As can be observed, the role of the articles varies: from **P-3**, which contributes to only one research item, to **P-4**, which can be regarded as forming the groundwork of this thesis. Moreover, the implementation of the first research objective occurs only in the middle of the timeline, highlighting the importance of the articles collectively contributing to fulfilling the prerequisites. Finally, the order of addressing the research objectives reflects **a progression from the simplest to the most complex methods for maximising influence in multilayer networks**.

2.3.1 Research prerequisite 1

Prerequisite: *Identify various methodologies for modelling spreading phenomena in multilayer networks, and select a single mechanism under which all research objectives will be evaluated.*

Scientific articles:

- [P-1] *Simulating Spreading of Multiple Interacting Processes in Complex Networks,*
- [P-2] *Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks,*
- [P-4] *Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model.*

Details: Addressing this prerequisite was undertaken in the first two articles chronologically (**P-1**, **P-2**) and complemented in **P-4**. In total, two main approaches were proposed and released in the form of a computational package.

Following the taxonomy for coexisting spreading phenomena [41], a framework capable of modelling them was introduced [P-1]. Its principal assumption was to combine a set of spreading processes into a single model, under the condition that they consist of discrete states (Def. 1.2.4). The method represents these processes on a hyperdimensional grid, whose vertices

are obtained as the Cartesian product of the states of the modelled phenomena, with edges denoting allowed transitions. For instance, the Susceptible-Infected-Recovered (SIR) model combined with the Unaware-Aware (UA) model is expressed as a six-element grid (SU, SA, IU, IA, RU, RA), where the transition $SU \rightarrow IU$ represents an unaware individual contracting the infection. By manipulating the transition weights, this model can be calibrated to simulate relations between processes, such as support or suppression. Employing this methodology enables the simulation of spreading phenomena in multilayer networks by associating each process with a network layer, i.e. one relationship may serve as the medium for disseminating information A, while another conveys information B.

The second approach to modelling spreading processes focuses on a single phenomenon that propagates across all layers of the multilayer network. Building upon [42], which considers only multiplex networks (i.e. a special case of multilayer networks where all actors are present in every layer), the mechanism proposed therein was relaxed to encompass arbitrary multilayer networks [P-2]. The method introduces an additional fusion component on top of the layer-specific spreading models, namely the protocol function (Def. 2.3.1). This function, parametrised by a threshold value, governs how to aggregate node-level inputs across layers for each actor in order to determine its final state. If the threshold is exceeded, the actor and all its corresponding nodes are activated; otherwise, it remains inactive, even if some of its corresponding nodes have received positive input.

Definition 2.3.1 (Protocol Function). *According to [42], state of the actor a of the multilayer network $M = (A, L, V, E)$ in the time step t is determined by a function:*

$$y_a(t) = \begin{cases} 1, & \text{if } x_a(t) \geq \delta \text{ or } y_a(t-1) = 1 \\ 0, & \text{otherwise} \end{cases}$$

Where:

$x_a(t)$ is the aggregated state of all nodes representing the actor,

$\delta \in [\frac{1}{L}, 1]$ is a parameter.

The main benefit of employing such an approach lies in its universal design, which allows for extending other models, e.g., the Linear Threshold or the Independent Cascade. Its drawback, however, is that an influence function built upon such a model ceases to be submodular [P-4], meaning i.a., that the greedy seed selection heuristic no longer yields results with an approximation guarantee. This, however, is a trade-off that must be accepted when employing a more complex network model, like in the problem considered here.

In the following research activities, the latter of the two proposed approaches was adopted, as it aligns more closely with the canonical formulation of the influence maximisation problem [28]. Moreover, whether the study considers LT- or IC-based models, two extreme cases of the protocol function were analysed: the rigorous case $\delta = 1$ (hereinafter referred to as *AND*) and the permissive case $\delta = \frac{1}{L}$ (hereinafter referred to as *OR*). In the former, an actor is activated

only if it receives sufficient influence on all layers in which it is represented; conversely, in the latter, receiving sufficient input in at least one layer is enough for activation.

Another, equally important, contribution under this prerequisite was addressing the gap in research environments for analysing multilayer networks and simulating spreading phenomena. To this end, a Python library named **network-diffusion** [P-1, P-2] was designed, developed, and published, consolidating the implementations of the methods described above. The environment was intended to meet both scientific and software engineering standards, namely to be (1) scalable, (2) accessible, (3) compatible with other data and network science toolkits, and (4) reliable. The first requirement emphasised extensibility, given the long research horizon during which new methods and approaches might emerge. Accessibility aimed to facilitate reuse in subsequent studies and to make the library available to the broader network science community. Compatibility was intended to improve usability, while reliability ensured that results obtained with the framework are reproducible.

As a result, **network-diffusion** is capable of simulating spreading phenomena in both temporal and multilayer networks. It is equipped with numerous pre-defined spreading models as well as open interfaces that allow new models to be implemented quickly. The library is publicly available (<https://network-diffusion.readthedocs.io>), easy to install, thoroughly tested (coverage ca. 90%), and well documented, making it accessible to other research centres. Moreover, since the framework has served as the backbone for all works covered in this dissertation, it has been extended several times with new concepts and methods developed in the process of addressing the research objectives. For instance, it has been augmented with centrality metrics and seed selection methods for multilayer networks [P-4] and with algorithms for obtaining the Minimal Dominating Set [P-5].

2.3.2 Research prerequisite 2

Prerequisite: *Develop metrics for quantifying spreading effectiveness in multilayer networks, enabling comparison of spreads triggered under varying conditions.*

Scientific articles:

- [P-4] *Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model.*
- [P-5] *Applicability of the Minimal Dominating Set for Influence Maximization in Multilayer Networks,*
- [P-6] *Identifying Super Spreaders in Multilayer Networks,*

Details: In order to address the observed gap in available metrics, four approaches have been proposed. As this task represents a prerequisite for the core problems examined in this dissertation, the work concerning it has been conducted largely in parallel with the main

objectives, and its outcomes are therefore distributed across three articles. In **P-4** and **P-5**, metrics for assessing the quality of a seed set are introduced, while in **P-6**, where the problem of influence maximisation had to be approached from a different perspective, a methodology for evaluating the “spreading quality” of a single actor was devised.

Definition 2.3.2 (Gain [P-4]). *The notion of the final range of the phenomenon being propagated across the network M w.r.t. the given seed set is captured by Gain:*

$$\Gamma = \frac{|S_\infty - S_0|}{|A - S_0|} : \Gamma \in [0, 1] \quad (2.1)$$

Where:

A — the set of all network’s actors,

S_0 — the seed set,

S_∞ — the set of active actors at the steady state.

Definition 2.3.3 (Area under Curve [P-5]). *The notion of the diffusion dynamics Λ is captured by measuring the area under a normalised activation curve:*

$$\Lambda = \int_0^1 Y(M, t) dt : \Lambda \in [0, 1] \quad (2.2)$$

Where:

M — the multilayer network,

$t \in (0, 1]$ — the normalised simulation step,

$Y(M, t) = \frac{1}{|A|} \sum_{a \in A} y_a(t)$ — the normalised activation curve.

These two metrics are related, as they both consider a single instance of diffusion. The higher their values, the more effective the given spread is. It is worth noting that they abstract from both the network size and the seed set size; thus, employing them in experiments enables the generalisation of conclusions. Nonetheless, they capture two distinct properties: a spread with a greater Γ does not necessarily exhibit stronger dynamics (Λ).

When the effectiveness of a spread is decoupled from the particular network and seeding budget, it becomes possible to develop additional metrics. This was the case in [P-4], where, in order to assess influence maximisation methods, the concept of effectiveness curves was introduced. The purpose of this approach is to establish a general relationship between the spreading regime and Γ . This enables the comparison of the expected diffusion strength triggered by different seed selection strategies.

Fig. 2.1 presents a schematic illustration of the procedure. Obtained values of Γ are plotted as a function of two parameters influencing diffusion: the activation threshold (vertical) and the seeding budget (horizontal) — the greener the tile, the higher the gain. By subsequently applying a gradient operation and binarisation, the heatmap is partitioned into three regions

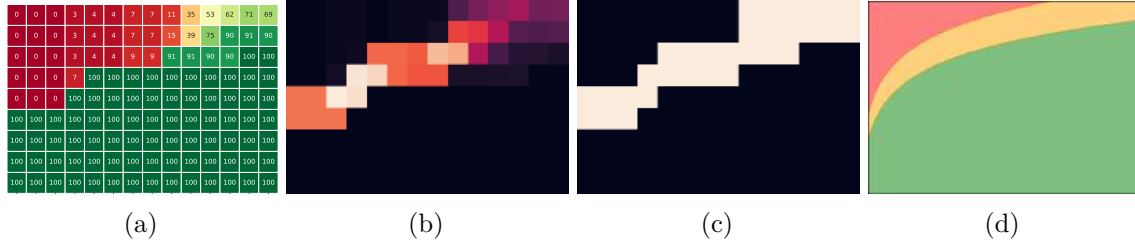


Figure 2.1: An Example of obtaining efficiency curves. Fig. 2.1a shows a heatmap of the raw Γ , Fig. 2.1b depicts the computed gradient magnitude, Fig. 2.1c shows the thresholded gradient that partitions the area into three sections, and Fig. 2.1d presents the two resulting curves: the bottom one separates the efficient and transitional regions, while the upper one delineates the transitional-inefficient border.

reflecting diffusion power: effective (green, where the process reaches $\Gamma \approx 1$), ineffective (red, where spreading does not occur), and transitional (yellow, where Γ is neither large nor close to 0). Finally, a curve-fitting procedure is applied to derive analytical expressions for the functions that separate these regions.

With Def. 2.3.4, the notion of spreading capabilities for individual actors has been introduced, extending the ideas of [90]. As can be observed, the vector captures four distinct perspectives on the question of what makes an agent an effective spreader. Furthermore, the relative importance of its components may vary depending on the specific problem. For instance, in epidemic suppression, p_{pi} might be more crucial, whereas in a pro-reading campaign, p_{ex} could play a greater role.

Definition 2.3.4 (Spreading-potential vector **P-6**). *A vector of spreading potentials can be defined for each actor from the network M as follows:*

$$\mathbf{p} = [p_{ex}, p_{sl}, p_{pi}, p_{pl}]$$

where:

- p_{ex} - total number of activated actors across the network,
- p_{sl} - spreading duration, as a number of spreading steps,
- p_{pi} - maximum number of actors activated simultaneously,
- p_{pl} - iteration at which the activation peak occurred.

2.3.3 Research prerequisite 3

Prerequisite: *Provide multirelational data that can serve as a structural backbone for experiments on influence maximisation and develop methods capable of real-time augmentation.*

Scientific articles:

- **[P-3]** *Peer Interaction Dynamics and Second Language Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social*

Network Analysis,

- [P-4] *Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model.*
- [P-7] *Multilayer Artificial Benchmark for Community Detection.*

Details: Articles associated with this prerequisite contribute to it in different ways. Specifically, **P-3** and **P-4** focus on collecting a representative sample of networks to address **RO-1**, **RO-2**, and **RO-3**, whereas **P-7** made it possible to complete **RO-4**.

The first requirement for the experimental part was to obtain a set of multilayer networks to serve as a environment for simulations. Since the selected spreading mechanism (see Sec. 2.3.1) does not rely on edge, actor, or node features, the sample was chosen to represent various systems (biased towards social ones), with an emphasis on their structural properties. To this end, several multilayer networks capturing diverse scenarios were included: scientific co-authorship [91], social interactions between academics within a single department [59], innovation diffusion among physicians [58], relationships within a corporation [60], air transportation networks [10], and interactions between users of a “virtual world platform” for young teenagers [9]. These were supplemented with two popular synthetic models: Erdős–Rényi [92] and scale-free [93], generated using [55] with their extensions to the multilayer setting. This ensured that experiments conducted with this dataset could also be interpreted in reference to the most prominent synthetic models.

In addition to the networks described above, which were obtained from studies conducted by other academics, an original one was also included. The **12-course** network was created as a contribution to a research project in which the author of this dissertation was involved [P-3]. The study aimed to examine the effectiveness of second-language acquisition during a three-month abroad course involving ca. 40 US adolescents in Jordan. To measure learning trends among the students, the social network analysis paradigm was employed, and therefore the raw data collected by linguists and psychologists had to be transformed into a graph-like representation. The resulting network made it possible to identify which social interactions facilitate foreign-language learning, and its multilayer variant has been included in the core dataset used in this research. It is worth noting that the value of **12-course** lies in its context — it was built upon real encounters between people, which makes it particularly important in the era of networks scraped from the Internet.

Finally, the last contribution within this prerequisite was the design and implementation of **mABCD** — a synthetic multilayer network generator . As stated in [P-7] Chp. 1, the available methods are either simple yet fast or offer fine-grained configurability at the expense of efficiency. To bridge this gap, a new approach to generating synthetic multilayer networks was proposed. This allowed the last research question to be addressed.

The main design assumption was that, since multilayer networks are not merely collections

of graphs independently spanned over the same set of nodes, the generator must be capable of modelling these complex interlayer relations. Consequently, the operating principle of **mABCD** was determined only after a structural analysis of the networks included in the aforementioned dataset. The outcomes of this examination revealed no consistent pattern in the relations between network layers. Therefore, the proposed generator was designed to provide users with the flexibility to define the expected geometry of the network to be generated. Moreover, the framework was based on the well-known **ABCD** [94], thereby inheriting its most prominent features, such as scale-free distributions of community size and degree, adding to that the ability to capture interlayer dependencies. The implementation was developed in Julia to ensure computational efficiency and backward compatibility with **ABCD**, but ports to Python are also available, thus the integration with **network-diffusion**. It is worth noting that, the generator was developed following best software engineering practices to ensure reliability, efficiency, and extensibility. It is also publicly available to facilitate use by other researchers (<https://krainskil.github.io/MLNABCDGraphGenerator.jl>).

2.3.4 Research objective 1

Objective: *Evaluate various rank refining methods for influence maximisation in multilayer networks and assess how they differ in terms of spreading effectiveness.*

Scientific articles:

- [P-4] *Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model.*

Details: The work related to this research objective evaluates various rank-refining seed selection methods for the budget-constrained influence maximisation problem in multilayer networks under the multilayer linear threshold model. In total, **ten methods** belonging to this category have been considered, with random and greedy choices serving respectively as the lower and upper bounds. Furthermore, as this dissertation assumes actor-based spreading (following the outcomes of **RP-1**), several of the employed methods were extended to this scenario in order to meet the research scope.

For the heuristics based on network-centrality metrics, five algorithms were evaluated: *Degree Centrality*, *Neighbourhood Size*, *K-shell Decomposition* [95], *PageRank* [96], and *VoteRank* [97]. The three latter metrics required redefinition from the domain of monoplex graphs to the multilayer setting.

As this extension constitutes a contribution beyond the influence maximisation problem itself, the methodology is presented here in a concise form. The modifications were implemented in two ways. The first, “naïve” approach assumed that the value of a given metric for a particular actor was the average of the values computed for all nodes representing this actor

across the network layers, using the canonical variant of the metric for each node. For instance, the mean of the PageRank scores for all nodes associated with a given actor was taken as that actor’s PageRank value. The second approach involved redefining the metrics to operate on actor neighbourhoods rather than node degrees, thereby obtaining these metrics directly for actors.

The next group of evaluated methods comprised *Community-Based Influence Maximisation* [43], *Clique Influence Maximisation* [98], and *K++ Shell* [68], which were originally defined for multilayer networks but were adapted here to the actor-based spreading models. In addition, *Degree Centrality Discount* [99] was employed, and, inspired by this approach, a new heuristic — *Neighbourhood Size Discount* — was introduced as a novel seed selection method.

Regarding the methodology of evaluation, the experiments were conducted in two stages across a diverse space of spreading regimes and networks. The first phase involved ten small networks, whereas the second was focused on two large datasets comprising tens of thousands of actors. For both stages, the parameter space of the propagation model was identical yet comprehensive, capturing a wide range of conditions and Γ metric was used to assess diffusion effectiveness. This two-stage design aimed first to identify the most promising approaches amongst those employed and then to evaluate only these on the large-scale networks.

The preliminary assessment on the ten smaller networks revealed that the five most effective methods were *Degree Centrality Discount*, *Neighbourhood Size Discount*, *PageRank* (variant *p-rnk*), and *VoteRank* (both variants: *v-rnk* and *v-rnk-m*). In the second stage, *v-rnk-m* emerged as the most effective method. However, due to its high computational complexity, it was not considered in subsequent studies. Instead, the second and third performing approaches, *Degree Centrality Discount* and *Neighbourhood Size Discount*, were employed, as they yielded results of similar quality.

The work concludes that all evaluated parameters — including the internal parameters of the spreading model, the network type, the budget, and the seed selection method — collectively determine the overall effectiveness of influence maximisation. Moreover, there is no single seed selection method that consistently yields the best results. Apart from random choice and the greedy approach, which form clear lower and upper performance bounds, the mean performance of the evaluated methods is largely levelled under the *OR* protocol. Interestingly, this phenomenon is not observed under the more rigorous *AND* protocol, where the mean Γ values are more widely dispersed across methods. This suggests that, in real-world scenarios such as opinion propagation, it may be more effective to focus on reducing activation thresholds (e.g. through mass propaganda or promoting desired attitudes via mass media) than on identifying a precise set of “influencers”. Finally, when the performance of methods is relatively close, computational complexity may serve as a secondary criterion: a computationally efficient algorithm (e.g. lightweight *Degree Centrality*) may be preferred over a more expensive one (e.g. *PageRank*) if the difference in effectiveness is marginal.

2.3.5 Research objective 2

Objective: *Propose a seed-selection approach that leverages the minimal dominating set for the influence maximisation problem in multilayer networks and verify its effectiveness.*

Scientific articles:

- [P-2] *Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks,*
- [P-5] *Applicability of the Minimal Dominating Set for Influence Maximization in Multilayer Networks.*

Details: In order to conduct this study, the concept of the minimal dominating set (MDS), first extended to the actor-based multilayer scenario in [P-2], was employed. This concept plays a key role in the domain of network controllability [30], as the entities it comprises (in the case of multilayer networks, the actors) are adjacent to all remaining agents in the given system. By minimising size of such a set, one can obtain a small number of actors capable of directly influencing the entire network. Formally, these concepts are defined as follows:

Definition 2.3.5 (Dominating Set). *A set of actors D' dominates the multilayer network M if for every actor $a \in A$, either a belongs to D' , or for each layer where a is represented, the corresponding node of a is adjacent to a node that represents an actor in D' .*

Definition 2.3.6 (Minimal Dominating Set). *A set of actors $\hat{D}$ is called a minimal dominating set if it satisfies Def. 2.3.5 and none of its proper subsets is a dominating set.*

It is worth noting that, as a result of the preliminary analysis, this work proposes and employs an improved algorithm for finding an MDS, inspired by [100]. The results obtained show that it can produce solutions up to 63% smaller than those generated by the baseline greedy heuristic [P-2].

As the goal of this objective was to verify the applicability of the MDS in the influence maximisation problem, it was necessary to develop a methodology to assess this in an objective manner. Using the MDS directly was not feasible, as it is prone to combinatorial effects. Also, in many cases, it can exceed the assumed seeding budget. Consequently, to address these issues the leading heuristics from [P-4] were employed in the study in two variants: the original implementation and a modified version that includes an additional filtering step, preserving only those actors belonging to the MDS. In this way, it was possible to reach the research objective by comparing the diffusion processes triggered by respective seed sets selected with and without considering the MDS.

The spreading phenomenon was again simulated using the multilayer linear threshold model. Regarding the datasets, the study considered Erdős-Rényi, scale-free, and real-world

networks. Diffusion effectiveness was measured using Γ and Λ , as preliminary experiments indicated that the former alone was insufficient for a comprehensive comparison.

The results reveal that utilising the MDS substantially alters the seed sets, which share on average only 36% similarity with those obtained using the baseline methods. The diffusion boost depends on the spreading regime and is particularly pronounced under low layer-level activation thresholds (up to $\mu = 0.4$) and the rigorous protocol function ($\delta = AND$), regardless of network type. In real-world scenarios, this spreading regime may correspond to situations in which convincing someone requires fewer acquaintances, but these connections must span all layers.⁴ Higher seeding budgets, where the ratio in the MDS of actors used as seeds grows, further amplify this effect. The underlying reason is that, when the MDS is considered in the seed selection process, seeds are distributed more widely across the network, including its periphery, whereas relying on baseline methods results in seed sets concentrated in the network core. At lower activation thresholds, this mechanism ensures faster coverage of the entire network. Conversely, as the threshold increases, MDS-backed methods lose their advantage sooner, as their influence is divided between the core and the periphery.

It was also observed that scale-free networks benefit the most from incorporating controllability concepts into the seed selection process ($\Delta\Gamma$ up to 0.26), owing to their topology. Unlike Erdős-Rényi networks, which distribute connections more evenly, scale-free networks gain from seed sets that cover both the core and the periphery, making MDS-backed methods more effective in triggering diffusion. By contrast, Erdős-Rényi networks are less suitable for MDS-based approaches because their degree distribution results in an MDS that is more randomised and typically too small to satisfy the seeding budget. The results for real-world networks lie between those of the synthetic models. Nonetheless, for these networks, incorporating the MDS yields also the greatest benefit under the *AND* protocol with lower activation thresholds. However, the example of the *arxiv* network demonstrates that, in some cases, using the MDS is ineffective when it covers the entire set of actors.

2.3.6 Research objective 3

Objective: *Develop an inductive graph neural network that predicts which actors in a given multilayer network are the most effective spreading seeds, and validate its performance against other state-of-the-art methods.*

Scientific articles:

- [P-6] *Identifying Super Spreaders in Multilayer Networks.*

⁴For instance, consider a presidential election with highly polarised candidates. We may not be aware of the opinions of all our acquaintances, either because some prefer not to share them or because we choose not to discuss politics. Nevertheless, we often know the views of a few individuals within each of our social circles, and if the majority of them hold the same opinion, we are more inclined to adopt it as our own.

Details: Completing this objective was not a straightforward process, as deep-learning approaches in this area are not widely used. Therefore, it was first necessary to establish a paradigm under which the task would be solved. Unlike **RO-1** and **RO-2**, where the influence maximisation problem was considered in its canonical form (e.g. budget-constrained), applying it directly here would have been too cumbersome. This is due to the limitations of the learning task: predicting seed sets leaves many degrees of freedom (their size can range from a single actor to the entire network), and obtaining ground-truth data is nearly impossible because of the combinatorial complexity.

These issues were addressed in **[P-6]** by formulating the problem of identifying super-spreaders, defined as those actors who, when used as a single source of diffusion, make the spreading most effective. Nevertheless, as already noted, diffusion effectiveness can be understood in several ways, particularly when triggered from a single source. From the other hand, framing the problem as a binary question (whether a given actor is or is not a super-spreader) risks oversimplifying it.

Therefore, **[P-6]** considers a feature vector $\mathbf{p}$ associated with each actor in a given network. Each component of $\mathbf{p}$ reflects a specific perspective on the actor’s “spreading potential” (Def. 2.3.4) and is obtained through diffusion experiments that employ the Multilayer Independent Cascade Model, as applying the Linear Threshold mechanism is not feasible in this setting. Building on this concept, the study introduces a “spreading potential score” y as a scalar representation of $\mathbf{p}$, which enables the formulation of the super-spreader identification task as the prediction of a ranking $\mathbf{R}$ of actors sorted w.r.t. y .

Another challenge addressed to achieve this objective was how to measure the quality of $\hat{\mathbf{R}}$, as it was observed that the distribution of y is highly biased and depends on both the spreading regime and the network under consideration. Consequently, instead of using standard metrics such as *precision@k*, the task of identifying the top- k super-spreaders was defined as follows:

$$\phi(M) \rightarrow \hat{\mathbf{R}} : \min \left| 1 - y_{\text{rel}}(\mathbf{R}, \hat{\mathbf{R}}, k) \right| \quad (2.3)$$

That is, the study sought a model ϕ which, given a multilayer network M , produces a ranking $\hat{\mathbf{R}}$ such that the cumulative spreading potential score of the identified top- k super-spreaders, relative to that of the ground truth (y_{rel}), is the most accurate.

As a result, a novel GNN model — *TopSpreadersNetwork* — was proposed. It accepts a multilayer network as input and outputs, for every actor, a vector $\hat{\mathbf{p}}$ from which a ranking $\hat{\mathbf{R}}$ is further constructed. It is worth noting that, by yielding $\hat{\mathbf{p}}$ as an intermediate output, the model gains additional flexibility. In particular, it can be employed to select diffusion seeds when the objective differs from simply maximising the total number of activations generated by the diffusion process. Another advantage of the *TopSpreadersNetwork* lies in its inductive design, which enables inference on previously unseen data. In summary, the

model performs prediction in three consecutive steps: (1) each layer of the input network is processed independently by a shared encoder that produces node-level embeddings, (2) these embeddings are aggregated by a trainable fusion component to obtain actor-level embeddings, and (3) a head component transforms them into vectors $\hat{\mathbf{p}}$. Thanks to this design and the sequential processing of actors' local neighbourhoods, the model is independent of network size, both in terms of the number of layers and the number of actors.

In order to evaluate its generalisation capabilities, the dataset described in **RP-3** was extended with two hundred artificial networks generated using the `multinet` library [55] (as the generator mentioned in **RP-3** was not yet finished). The training process relied exclusively on synthetic graphs, whereas the evaluation was conducted on real-world networks. Finally, the developed model was compared with competitive solutions representing a range of approaches to the influence maximisation problem: network centralities (Degree Centrality, Neighbourhood Size), rank-refining heuristics (Degree Centrality Discount [99], Neighbourhood Size Discount [**P-4**]), unsupervised learning (Multi-node2vec [11] with KMeans clustering), and reinforcement learning (DeepIM [78]).

The results reveal that *TopSpreadersNetwork* achieved the best performance in 7 out of 8 k -related scenarios, significantly outperforming all other methods, particularly for the topmost spreader, which highlights its robustness. Amongst the baselines, only the degree-based methods delivered comparable results, while all others attained substantially lower scores. The model's performance was especially pronounced on the *timik* [9] network, demonstrating its effectiveness on large-scale data. Nevertheless, a performance drop was observed on networks with considerably more layers than the maximum of 5 in the training data. This may result from limitations of the aggregation component and could be addressed using techniques analogous to those employed for handling large neighbourhoods.

2.3.7 Research objective 4

Objective: *Design a method to manipulate the structure of a multilayer network at a macro scale, enabling comparison of spreading effectiveness between the original and modified networks, and identify the structural properties that facilitate the influence maximisation process.*

Scientific articles:

- [**P-7**] *Multilayer Artificial Benchmark for Community Detection (mABCD).*

Details: This work, conducted in parallel with the development of the `mABCD` graph generator, proposes a preliminary approach to controlling diffusion in multilayer networks by manipulating their structure. In this way, the influence maximisation problem is reframed as a special instance of spread control, and by leveraging this approach, the research objective is addressed.

The methodology assumes that the analysed real-world multirelational system is encoded into a configuration file for a synthetic graph generator, which is then used for further modelling. By modifying selected parameters of the generator, one can explore a variety of “what-if” scenarios and construct adjusted “digital twins” that reflect possible interventions on the original system. These can be evaluated under the assumed spreading regime, enabling assessment of which topology is optimal for a given influence control task (in this case, maximisation). Leveraging **mABCD** as a generator of synthetic multilayer networks is crucial, as it provides both high configurability and computational efficiency, a combination that is unique among competing tools. Nonetheless, other frameworks could be employed, with the caveat that this component lies at the core of the pipeline, as its parameter space must correspond with structure of the analysed system.

In summary, the proposed approach comprises the following steps:

1. select a real-world network (that constitutes the subject of the experiment);
2. derive a set of configuration parameters for the given graph generator w.r.t. the network;
3. generate a set of the network instances serving as “digital twins” of the real-world system;
4. simulate diffusion under a predefined regime;
5. manipulate a specific parameter of the configuration model;
6. generate a set of the network instances based on the modified configuration;
7. simulate diffusion under the same regime;
8. compare the diffusion dynamics and assess the influence of the analysed parameter.

The study evaluates three parameters in the area of spreading phenomena analysis: (1) the noise level between communities, (2) the number of actors comprising the network, and (3) the correlations between communities across layers. The Multilayer Independent Cascade Model is utilised as the spreading mechanism, which stems from the constraints imposed by employing **mABCD**. In brief, the graph generation process is designed to capture community structure, i.e., global rather than local properties. Consequently, as expected, the generated graphs contain groups of nodes inducing denser subgraphs relative to the global density, but locally neighbourhoods still resemble tree structures, as there are very few short cycles. In such a structure, applying derivatives of the Linear Threshold Model is infeasible, as diffusion unfolds abruptly: it either saturates the network immediately or fails to trigger at all.

It is worth noting that the conducted experiments serve only as a proof of concept for the devised approach, as its inherent complexity exceeds the scope of this dissertation. This is primarily due to the need for appropriate metrics for comparing multilayer networks which, given the nondeterministic behaviour of **mABCD**, hinders control over the generated “twins” and thereby affects the stability of the experiments. Nevertheless, the results obtained in the study confirm that manipulating selected topological parameters enables control over diffusion, although each parameter contributes in a distinct manner.

Regarding the inter-community noise level, its impact was found to vary significantly between the two diffusion regimes (*OR* and *AND*). Under the more restrictive protocol

function, increasing inter-community noise enhances information spread, with the effect becoming even more pronounced when noise is heavily suppressed: in such cases, diffusion coverage may decrease by up to 25%. In contrast, when the *OR* protocol is applied, the effect is reversed — reducing noise improves spreading — although the magnitude of this effect is considerably smaller.

Interestingly, the results reveal that preserving all configurational parameters except for the number of actors affects spreading only when this number is increased. Decreasing the number of actors while reusing the remaining parameters retrieved from the real network does not alter the diffusion behaviour. Conversely, when the network is upsampled, slight but non-negligible changes in diffusion dynamics are observed: the more the network is upsampled, the larger the change in Γ . This observation provides an interesting basis for further research on the role of graph reduction in the analysis of large-scale networks.

Finally, the strength of correlation between groups appears to have little influence on information spread. This parameter therefore seems of limited use for controlling diffusion.

Altogether, the findings confirm the feasibility of the proposed approach to controlling spreading in multilayer networks. However, it is worth noting that some follow-up work related to this research objective has already been undertaken, although it lies beyond the scope of this dissertation. Building on the findings described above, a study aimed at closing research gaps in the problem of finding “digital twins” for multirelational systems is currently ongoing, where the configuration retrieval task with **mABCD** is getting further developed.

Chapter 3

Conclusions

The research comprising this dissertation explored the domain of spreading phenomena in multilayer networks from the perspective of the influence maximisation task. Based on an extensive literature review, four scientific problems were identified and formulated as research objectives. These were accompanied by three prerequisites that had to be addressed in order to proceed with the core part of the study. In total, all seven work items have been successfully completed. The outcomes were presented in seven scientific articles, published either in reputable journals or as proceedings of highly regarded conferences.

In particular, the dissertation can be summarised by the following key findings.

- Rank-refining methods for influence maximisation in multilayer networks, despite residual differences between them, yield broadly similar results for a given spreading scenario. Together with other parameters (i.e. the seed set budget and the diffusion model), they jointly contribute to spreading effectiveness. Hence, in real-world scenarios it is recommended not to limit the analysis solely to the choice of the seed selection method (from **RO-1**).
- The proposed method of incorporating the minimal dominating set into the seed selection process can intensively enhance diffusion, but under specific circumstances, i.e., when the network in which information propagates follows a scale-free model, and the spreading regime assumes a non-restrictive node activation condition combined with a rigorous aggregation approach across layers for determining the final state of an actor (from **RO-2**).
- The feed-forward architecture of the graph neural network devised in this research proves effective in identifying super-spreaders in multilayer networks. Moreover, the proposed design exerts generalisation capabilities, which makes it more robust compared to other deep learning-based methods (from **RO-3**).
- Spreading phenomena in multilayer networks can be effectively controlled through structural interventions. Nonetheless, the impact of modifying a given geometrical

property depends on the spreading regime, and may even be inverted with abrupt changes in the parameters of the spreading model (from **RO-4**).

Furthermore, the work undertaken whilst addressing the research objectives yielded several contributions. Foremost, a new seed selection method, *Neighbourhood Size Discount*, was proposed, and centrality algorithms — *VoteRank*, *PageRank*, and *K-shell Decomposition* — were adapted to the multilayer setting. Within the field of network controllability, the concept of the minimal dominating set was extended to the actor-centric scenario in multilayer networks, and an algorithm for its identification was proposed, and a method for utilising the minimal dominating set in the seed selection process was also introduced. Another contribution was the proposal of an inductive graph neural network architecture for identifying super-spreaders in multilayer networks of arbitrary size and relational complexity — *TopSpreadersNetwork*. The architecture was accompanied by an extensive dataset illustrating the spreading capabilities of actors across various systems. Finally, a method for projecting real-world multilayer networks onto a set of configuration parameters of a synthetic graph generator was proposed.

Since the research objectives were preceded by the prerequisites (**RP-1**, **RP-2**, **RP-3**), several additional outcomes were also obtained. In particular, a methodology for modelling coexisting and interacting spreading phenomena was proposed, several metrics for assessing diffusion effectiveness from different perspectives were defined, and a real-world dataset illustrating social connections between students in an isolated overseas environment was constructed. Moreover, the Python package `network-diffusion` was developed to facilitate the analysis and simulation of spreading phenomena, and an effective method for generating scale-free synthetic multilayer networks was proposed and implemented in the form of the computational library `mABCD`.

Based on the obtained results, it can be concluded that the overarching goal of this dissertation has been successfully achieved. The findings demonstrate that influence maximisation in multilayer networks is inherently dependent on multiple factors. By utilising the perspective of Information and Communication Technology, the problem can be effectively addressed by approaches leveraging complementary paradigms, each offering distinct advantages and limitations (as noted above).

The author firmly believes that the research presented in this dissertation provides a comprehensive answer to the scientific problem as it was formulated and addressed herein. Yet, as with any matter of inherent complexity, the issue could have been approached from multiple perspectives. Moreover, the findings reported in this work entail possible follow-up studies. Briefly highlighting these two aspects not only offers the reader a broader outlook, but also situates the undertaken research within the wider landscape of network science.

As for the first issue, Chp. 1 implicitly sheds light on “what could be done differently”; however, it is useful to summarise this aspect here. The most natural alternative path in addressing this problem would be to employ different network models to represent multirela-

tional systems, for instance by allowing hyper-edges [4]. Nevertheless, since the aim of this work was to contribute a subsequent building block to the state of knowledge in the field, multilayer networks naturally emerged as an extension of monoplex graphs, providing the additional dimension in terms of relation types. Another possible alternative perspective would be to explore different methodologies for tackling the influence maximisation problem in multilayer networks. Nevertheless, when determining the research scope, the author opted for the most promising approaches within each category, for example favouring feed-forward graph neural networks over recurrent architectures.

The second aspect — possible follow-up studies, should, in the author’s view, extend beyond the influence maximisation task and address the broader problem of spreading capabilities in real-world multirelational systems. To this end, future work can build extensively upon the last research objective realised in this study, tackling newly revealed challenges such as devising methods for measuring similarity between multilayer networks or developing coherent strategies for structural interventions in real-world systems by integrating the “digital twin” paradigm with synthetic graph generators.

Irrespective of the specific direction such studies may take, it will be interesting to observe further advances in understanding spreading dynamics in multirelational systems, as well as the development of novel methods emerging from this domain.

Chapter 4

Publications

The information preceding each article provides bibliometric details of the venue in which it was published, together with an objective account of the dissertation author's contribution. The underlying methodology is as follows.

MNiSW points indicate the “value” of the venue according to the scored list of journals and conference proceedings published by the Ministry of Science and Higher Education of the Republic of Poland (ascending scale from 0 to 200). The triennial CORE ranking denotes the category of a computer science and engineering conference as defined by the Computing Research and Education Association of Australasia (descending scale from A* to C). The IF refers to the Impact Factor, a journal-ranking metric calculated by the Web of Science (ascending numerical scale). Bibliometric data regarding MNiSW points and IF are provided in accordance with the DONA database (<https://dona.pwr.edu.pl/szukaj>) maintained by the Wrocław University of Science and Technology. The number of citations is reported with respect to the corresponding scholarly literature database. Finally, the CRediT (Contributor Roles Taxonomy) statement outlines the specific roles undertaken by each author using methodology developed by the Elsevier editorial group.

4.1 Simulating Spreading of Multiple Interacting Processes in Complex Networks

Authors: Michał Czuba, Piotr Bródka

Published at: 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA); October 2022

MNiSW points: 140 (Core A)

Citations: Web of Science: 3

(as of 6th March 2026) Scopus: 3

Google Scholar: 3

CRedit:

Conceptualisation

Methodology

Software

Validation

Formal analysis

Investigation

Writing – original draft

Writing – review & editing

Visualisation

Simulating Spreading of Multiple Interacting Processes in Complex Networks

1st Michał Czuba

Department of Artificial Intelligence
Wrocław University of Science and Technology
Wrocław, Poland
michal.czuba@pwr.edu.pl

2nd Piotr Bródka

Department of Artificial Intelligence
Wrocław University of Science and Technology
Wrocław, Poland
piotr.brodka@pwr.edu.pl

Abstract—Investigating the interaction between spreading processes in complex networks is one of the most important challenges in network science. However, whether we would like to know how the information campaign will affect virus spreading or how the advertising campaign of the new iPhone will affect the sales of Samsung phones, we need an environment that will allow us to evaluate under what conditions our spreading campaign will be effective. *Network Diffusion* is a Python package that should help do that. In this paper, we introduce its operating principle and main functionalities, including simple examples of simulations that can be performed using it.

Index Terms—interacting spreading processes, epidemics, network science, multilayer network

I. INTRODUCTION

For almost two years now, the resources of the entire world have been used for the fight against the "invisible" enemy. We spent countless hours and trillions of dollars to stop the COVID-19 pandemic, and still, millions of people have died from the virus or the collapse of healthcare systems around the world. One of the challenging tasks is understanding how different countermeasures, such as information campaigns, lockdowns, or vaccinations, affect the progression of the virus to choose those that will save both human lives and the economy.

In this paper, we would like to present the *Network Diffusion* package [1] that allows simulating the spread of many coexisting processes in single and multilayer networks. An example of such processes would be, of course, coronavirus spread, interacting with an information campaign and vaccinations, but also the interaction between two advertising campaigns (e.g. iPhone vs Samsung or Cannon vs Nikon), two political campaigns (e.g. Biden vs Trump), diseases (e.g., AIDS and Tuberculosis), to name just a few [2]. However, before we can present the *Network Diffusion* package, we would like to introduce a few key concepts from network science [3].

A. Complex networks

Complex networks are the backbone of all complex systems [3] starting from the nervous system of living organisms [4], through our infrastructure systems (electric grid, water pipes, airports, etc.) [5], [6], and ending with relationships

between people [7]. Complex networks can be represented and analysed in multiple ways, e.g., adjacency matrix [3] or property vectors [8], however, the most common approach is to represent a complex network as a graph [3] or a set of graphs (for multilayer networks) [9], [10]. The multilayer network, according to [9], is defined as quadruple $M = (N, L, V, E)$; where: N is a set of actors; L is a set of layers; V is a set of nodes, $V \subseteq N \times L$; E is a set of edges $(v_1, v_2) : v_1, v_2 \in V$, and if $v_1 = (n_1, l_1)$ and $v_2 = (n_2, l_2) \in E$ then $l_1 = l_2$.

B. Spreading phenomena

Spreading phenomenon covers a wide spectrum of processes, starting from information diffusion [11], through virus [12], opinion [13], innovation [14] spreading and ending with the spread of influence [15]. Fortunately, all these processes are similar if we look at their high-level components [3] (i) *What* phenomenon is spreading (e.g. information, opinion, virus), (ii) *How/Who* it is spreading (e.g. network nodes, actors, agents), and (iii) *Where* it is spreading (i.e. network type). All these elements carry a different context to the simulation results, but, at the same time, they can be imagined as "containers" for algorithmic structures, which will be introduced later. An example of those three components can be a viral video that spreads on a social networking site. Here, *what* is the video or digital content, *how/who* is a link to the website or a post on social media, and *where* is a social network that is the foundation of this particular social networking site.

1) *Epidemic modelling framework*: The initial models developed by epidemiologists are based on two assumptions [3]: compartmentalisation and homogeneous mixing. The first condition says that the state of each individual is discrete. This means (continuing with epidemiological analogies) that the relation of a given node to a disease can be described by n states, e.g. *healthy*, *sick*. The second assumption concerns the ability of the nodes to change their states. Each individual can interact with all nodes in a given time unit. Using that hypothesis, we can build many various spreading models like SI (Suspected-Infected), SIS (Suspected-Infected-Suspected), SIR (Suspected-Infected-Recovered), SIRS (Suspected-Infected-Recovered-Suspected), and so on. Epidemic models can be used to simulate other

This work was partially supported by the Polish National Science Centre, under Grant no. 2016/21/D/ST6/02408

spreading processes. The most common example would be information spreading, where we have models like UAU [16] (Unaware-Aware-Unaware) or UAF [17] (Unaware-Aware-Forgot), which are based on SIS and SIR models, respectively.

2) *Independent cascade model*: Independent cascade model (ICM) [15] has a much different mechanism and is widely used to simulate influence diffusion. Its principles reject the homogeneous mixing assumption because of the way the phenomenon propagates - it cannot be captured holistically for a given network. Here, the spreading takes the form of a cascade. Each newly activated node has one chance to activate its neighbours in each step. It is based on the likelihood of activation (aka. propagation probability) stored at the edge connecting them [18].

3) *Linear threshold model*: This model is based on the concept of the activation threshold that is defined for each node of the network [15]. It determines a minimum value of the influence of its neighbours to change its state. In other words, the minimum value above must be the sum of the intensity of the connections from the neighbouring activated nodes to result in activation [18]. When comparing ICM and Linear Threshold Model (LTM), one can say that the first one is based on the *push* mechanism, i.e., a node pushes its influence to its neighbours. In contrast, LTM is based on the *pull* mechanism, i.e. the node pulls the influence from its neighbours.

C. Interacting spreading processes in networks

When analysing how various phenomena in networks propagate, we have to ask the following questions: *what is being propagated?* and *where is it being propagated?*. As an answer, we get four general possibilities: (1) a single process in a single network, (2) a single process in a multilayer network, (3) multiple processes in a single network, and (4) multiple processes in a multilayer network.

The last two options bring the possibility of spreading multiple processes that can interact with each other. Thus, in addition to modelling each phenomenon, we also need to model the interactions between them, i.e., how they influence each other (and that is exactly a problem that *Network Diffusion* solves). Below, we briefly describe the four possible variants of interactions.

Firstly, we can distinguish supporting processes (about 11% of research in the domain [2]) that mutually support each other by increasing coverage and velocity. Here, a good illustration is that chronic diseases (such as asthma) are a catalyst for contracting COVID-19.

The next genres are competing processes (about 36% publications [2]). In this case, one process causes suppression of the propagation of the other. A good example of this is the presidential election: if the number of people influenced by candidate X increases, the number of people influenced by candidate Y has to decrease, and at the end of the day, only one candidate can win.

The third case, the mixed approach, covers about 46% of the publications [2]. It considers instances where one process supports the second, but the second competes with the first.

An interaction between disease and awareness can be a good example. People who get sick become aware of a disease, so the more people get sick, the more people will be aware. On the other hand, people aware of the pandemic will take preventive actions to limit its spreading.

The last case is when processes do not interact with each other. It appears only in 7% of the papers in the domain [2].

II. SIMILAR SOFTWARE PACKAGES

To gain valuable recognition of similar solutions to *Network Diffusion*, an appropriate tool review methodology was adopted. For this, the general state of the software available in the field was taken into account. We started from a simple reconnaissance using a search engine and focused more on the tools available for the Python language; however, it was not a strict condition. Our final goal was to see a general cross-section of state-of-the-art. As a result, information on more than twenty different tools was obtained. We analysed them and divided them into two groups: *packages dedicated for network spreading simulations* and *general complex network analysis software*.

A. Software for spreading processes simulations

1) *GLEaMviz*: The first application that has functionalities corresponding to the designed software is a GLEaMviz. It works with real data, population density, and migration around the world, combined with stochastic models of disease propagation. As a result, it provides a sophisticated simulation environment. Due to the large scale of the experiments (the whole world), a single node is a population of a given size (defined by the user). A very interesting feature is the manual definition of the epidemiological model. GLEaMviz makes this possible by manipulating the compartments (understood in the same way as in sec. I-B1). Allowable transitions between them are also fully definable. The user can also select the geographical start of the disease, the initial percentages of individuals belonging to a given compartment, its duration, etc. There is also an option to generate various visualisations at the end of the experiment. Despite the interesting functionalities mentioned above, GLEaMviz has a rather large disadvantage: it only allows the propagation of one process at a time [19].

2) *NDLIB*: Network Diffusion LIBrary is a Python package based on the NetworkX library. It allows performing simulations with many predefined epidemiological models (such as: SIS, SIR, SEIR, etc.), influence group (LTM, ICM, Profile, etc.), opinion group (Voter, Sznajd, etc.), and even dynamics (models with the capacity to change the topology of network). Moreover, the user can create its own customised models. Results visualisation is also possible via Matplotlib or Bokeh with the flexibility to append a custom graphical engine. NDLIB also has some interesting run-time features. First of all, it includes an option to perform a "multi-execution" of the simulation by parallel computing. As this kind of experiment is generally stochastic, this feature gives a chance to see the general behaviour of the observed phenomena. It also enables running the simulation on a server (as well

TABLE I
TOOLS WITH CORRESPONDING FUNCTIONALITIES TO *Network Diffusion*.

Name	Type	Environment	Functionalities	Access
GLEaMviz	app.	desktop	single model propagation, high freedom of model definition, visualisation of the experiment [19]	free
NDLIB	library	Python	single model propagation, high freedom of model definition, visualisation of the experiment [20]	free
SimInf	library	R	propagation of custom model, visualisation [21]	free
Sispread	app.	console	SI, SIS or SIR model propagation, returns numerical data [22]	free
STEM	Eclipse plugin	desktop	single model propagation, visualisation, workflow like finite element method (FEM) environments [23]	free

as locally). For users unfamiliar with Python, the authors created NDQL, a query language (based on SQL syntax) that supports elementary NDLIB commands. For those who cannot programme, they also provided a "visualisation framework" to play with some of the models implemented with a GUI-based tool [20]. NDLIB is a very useful library with many features. However, it does not directly support experiments where many processes interact together.

3) *SimInf*: The next software is SimInf, a process diffusion package for R. Its API has been designed in a very interesting way. This tool allows a user to define his own models by parsing the appropriate string. Interestingly, no network is needed to run the simulation, as the algorithms work on the assumption of homogeneous mixing. After a successful simulation, it is possible to display a summary graph. Nevertheless, we must note that no support for operations on real networks and the lack of an option to define multiprocessing experiments [21] are distinct shortcomings.

4) *Sispread*: Sispread is a simple application implemented in C. It is a console tool without a graphical interface. It focuses on three models (SI, SIS, SIR) with the possibility of deep analysis of the experiments performed. This tool supports very basic IO operations - the user can upload a custom network that meets certain requirements (without the option to manipulate it). As a result of the experiment, numerical data are returned for further analysis [22].

5) *STEM*: Another advanced system is STEM. As an extension of the well-known Eclipse IDE, it uses its graphical layout and the general philosophy of user interaction. It is reminiscent of FEM systems in the way it works. Just like them, it requires the user to specify a medium (equivalent to a model of a physical object) where the simulation is performed with its discretisation level (i.e. whether the node is a municipality or an entire county). The next step is to attach an appropriate solver propagation model and set the starting parameters of the experiment. Once this is done, it is possible to visualise the spread progress. As in other programmes, it does not support multiple spreading processes [23].

B. Software for complex network analysis

Due to the large number of tools found (see tab. II), we describe below only the most interesting of them.

1) *Gephi*: Gephi is an application for visualising networks in many aspects, with extensive graphical capabilities. It allows

the user to perform exploratory data analysis (EDA) in real-time on basic structures and connections between objects. Gephi is also designed for social network analysis - it has an easy way to map different portals (Facebook, Twitter, etc.) and Small World networks with each other. It can also visually represent biological data and serve as a tool for creating scientific posters. Additionally, within Gephi, the basic metrics for network analysis have also been implemented. The package works with many file types and supports integration with plugins written by external developers. In terms of non-functional aspects, it has an interface that does not require programming skills [27].

2) *iGraph*: iGraph is a library designed for several programming languages (i.e. C, R, Python, and Mathematica). The three main paradigms of iGraph are: easy handling of graph algorithms, fast handling of large networks, and enabling interaction with high-level languages like R. In many aspects, this tool is similar to pure NetworkX, but it is also able to run simple epidemic models like SIR. For clarity, it should be added that this functionality is rather residual [30].

3) *NetMiner*: NetMiner is a comprehensive environment for network analysis and visualisation. Unlike the tools described above, it is not free software. Among the functionalities that distinguish it from the others, we can list for sure the possibility of EDA (e.g., analysis of written text and, on this basis, creation of semantic network analysis), clustering, classification, etc. NetMiner also has extensive visualisation capabilities, both static and dynamic, and, what is interesting, three-dimensional. Another important feature of this tool is the presence of a Python-based console. It can even be upgraded by adding external packages. Moreover, NetMiner allows the compilation of written programmes, which is a big advantage of it in terms of securing the code against third parties [34].

4) *NetworkX*: NetworkX is one of the most important packages among Python data science libraries. Its capabilities are large and mainly concern static network analysis. NetworkX also has visualisation functionality, but in its official documentation, it recommends the use of Cytoscape, Gephi, and Graphviz to perform any visualisation [36]. Furthermore, because of its integration with the Python language, the long stability of this library can be assumed. NetworkX has a large number of algorithms and metrics that are useful for network analysis, which makes it virtually unrivalled among other Python programming tools. Moreover, this library is a

TABLE II
SOFTWARE FOR COMPLEX NETWORK ANALYSIS.

Name	Type	Environment	Functionalities	Access
AllegroGraph	DBMS	desktop	graph database management system, visualisation, static analysis of networks [24]	free
Arcgis	app.	desktop, web	static analysis of multilayer networks, visualisation, IO [25]	free
Cytoscape	app.	desktop	static analysis of networks, visualisation [26]	free
Gephi	app.	desktop	static analysis of networks, visualisation [27]	free
Giraph	library	Java	static analysis of networks [28]	free
Graphviz	app.	console	visualisation [29]	free
iGraph	library	Python, R, C, M	static analysis of multilayer networks, visualisation, IO [30]	free
Multinet	package	R Python	manipulation of multilayer networks, static analysis, visualisation IO [31]	free
Muxviz	app.	desktop	manipulation of multilayer networks, static analysis, static and dynamic visualisation [32]	free
Neo4j	app.	desktop	graph database management system, visualisation, static analysis of networks [33]	paid
NetMiner	app.	desktop	static analysis of networks, visualisation, IO, Python-like scripting [34]	paid
Network	library	R	static analysis of multilayer networks, IO [35]	free
NetworkX	library	Python	static analysis of networks, visualisation, IO [36]	free
NodeXL	plugin	Excel	static analysis of networks, visualisation, IO [37]	free, paid
Pymnet	library	Python	static analysis of multilayer networks, visualisation, IO [38]	free
Tulip	app.	desktop	static analysis of networks, visualisation, IO, Python-like scripting [39]	free
yEd	app.	desktop	static analysis of networks, visualisation, IO [40]	free, paid

base for several tools listed in this review.

C. Summary

Having the analysis of the existing solutions done, it is easy to see that they can be divided into several types based on the main functionality: (1) software for static analysis of networks, (2) software for visualising networks, and (3) software for simulation of process propagation in networks. We can also distinguish the following groups based on the type of the tool: (1) web applications, (2) locally executed applications, (3) libraries for languages popular in data science, and (4) others (database management systems, plugins for other applications, etc.). However, it is worth noting that, after careful reconnaissance, virtually no simulation environment was found for the simultaneous propagation of multiple processes.

III. OPERATING PRINCIPLE

The core functionality of the software requires adequate attention, especially since, as demonstrated in sec. II, no solution was found that covers the problem of spreading multiple processes in networks. Hence, before the start of designing the software, we had to address and answer two questions: (1) how to run any number of processes during a single experiment, and (2) how to determine (in a general way) the interactions between processes? To make further deliberations easier to understand, we defined a domain dictionary with all important terms (see tab. III)

A. Generalisation of single processes modelling

According to the concepts described in sec. I, we can notice a similarity between many discrete processes. Moreover, due to the numerical nature of the simulations carried out, it is not possible to hold the assumption of homogeneous mixing. Taking these into account, it is easy to conclude that models

like SIS, SIR, SIRS, etc. can be reduced to specific cases of a model with $T = [t_1, t_2, \dots, t_k]$ possible states, where each of the network nodes can be in one of them at a given time step. Moreover, the transition from one state to another takes place under a certain weight, which can be called pseudo-probability. Let us further note that, e.g. in the SIR model, it is impossible to transit directly from S to R . Thus, a state change is only possible between neighbouring states, i.e. from t_k one can only move to t_{k-1} and t_{k+1} .

Definition 3.1: It follows that every model $\mathfrak{M}$ belonging to the mentioned group can be described for the purposes of the designed system by: vector of states $T = [t_1, t_2, \dots, t_k]$ and vector of transition weights between neighbouring states $W = [w_1^2, w_2^3, \dots, w_{k-1}^k]$ (where w_n^m means a weight of transition $n \rightarrow m$); which can be combined in a matrix:

	t_1	t_2	t_3	$\dots$	t_k
t_1		w_1^2			w_1^k
t_2	w_2^1		w_2^3		
t_3		w_3^2			
$\vdots$				$\ddots$	
t_k	w_k^1				

For example, a SIRS model with coefficients: β (individual's infection rate), μ (individual's recovery rate), and ξ (loss of immunity) can be expressed as a vector of states: $T = [s, i, r]$; vector of transition weights between neighbouring states: $W = [\beta, \mu, \xi]$, and process matrix:

	s	i	r
s		β	
i			μ
r	ξ		

TABLE III
DOMAIN DICTIONARY FOR TERMS INTRODUCED TO DESIGN THE *Network Diffusion* PACKAGE.

Term	Symbol	Description
Epoch	$\varepsilon_i \in \varepsilon : \{1, \dots, \varepsilon\}$	A discrete-time unit of the Experiment when all Nodes of the Network have updated their State against each of the Processes.
Experiment	$X = (\mathfrak{M}, M, \varepsilon)$	A complete run of the Propagation model on the Network for a defined number of Epochs.
Network	$M = (N, L, V, E)$	A multilayer network (as introduced in sec. I-A).
Process	$T = [t_1, t_2, \dots, t_k]$	Singular phenomena affecting the Network, described by an ordered set of States. Each Process must consist of at least two States and, in the package, is identified with the particular Network Layer.
Propagation model	$\mathfrak{M} = (T_1 \times \dots \times T_n, W)$	A definition of how Processes should interact with each other, i.e. a set of all possible global States of the Nodes and a set of Weights under which Transitions are allowed during Experiment.
State	$t_{ajx} :$ $t_a \in T_1; t_j \in T_2; t_x \in T_3$	Categorical value denoting a relationship of the Node against a particular Process (a local State - here e.g. t_a) or a set of interacting Processes (a global State - here t_{ajx}).
Transition	$t_{aj} \rightarrow t_{ak} :$ $t_a \in T_1; t_j, t_k \in T_2$	Possible change of State of the Node at potential denoted by the Weight. Transitions are allowed between neighbouring States, e.g. they must differ only by one State.
Weight	$w_{t_j t_k}^k \in W$	Pseudo-probability of particular Transition occurrence (here: $t_j \rightarrow t_k$), valued by a number in range $[0, 1]$.

B. Example of two processes model

The way we defined the coexistence of several processes looks similar, but it is scaled into additional dimensions. As we know from sec. I-C, processes interactions can be supporting, competing, mixed, or independent. Let us consider a two-process model $\mathfrak{M} = (T_1 \times T_2, W)$ representing propagation of SIR-like disease (T_2) and UA-like vaccine against it (T_1) - a classic example of competing processes. Thus, we have:

- $T_1 = [u, v]$; u - unvaccinated individual, v - vaccinated individual
- $T_2 = [s, i, r]$; s - suspected individual (i.e. not yet affected by illness), i - infected individual, r - recovered individual

Both processes affect each node of the network. Therefore, we can describe all possible states in the model with the following set (a Cartesian product - $T_1 \times T_2$): $\{us, ui, ur, vs, vi, vr\}$. Moreover, under the assumption that at each simulation step, it is possible only to change the state of a node in the dimension of the singular process, we can list all available transitions (see fig. 1 for visual presentation):

- for T_1 (vaccination process); for s constant: $us \rightarrow vs$, for i constant: $ui \rightarrow vi$, for r constant: $ur \rightarrow vr$,
- for T_2 (disease process); for u constant: $us \rightarrow ui \rightarrow ur$, for v constant: $vs \rightarrow vi \rightarrow vr$

Please note, that if we were to assign appropriate weights to particular transitions that reflect the probability of a state change in one process given the others, we could obtain the effect of reinforcing or suppressing between processes. For example, $w_{us}^{ui} > w_{vs}^{vi}$ means that it is easier for unvaccinated individuals to get infected than for those who are vaccinated.

Moreover, with a more detailed analysis of fig. 1, we can see the advantages of the assumption that transitions are allowed only in the direction determined by the singular process. For our example, it means that changes of node's state can occur only in the T_1 axis (e.g. $us \rightarrow vs$) or the T_2 axis (e.g. $us \rightarrow ui$), but never "diagonally". This eliminates obviously incorrect transitions such as $ui \rightarrow vr$, which denotes a change

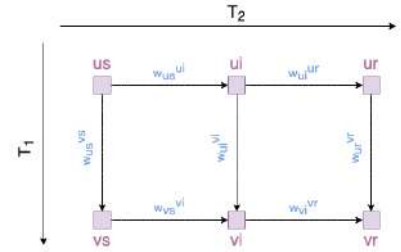


Fig. 1. Example of two process model $\mathfrak{M} = (T_1 \times T_2, W)$; $T_1 = [u, v]$; $T_2 = [s, i, r]$; $W = [w_{us}^{ui}, w_{ui}^{ur}, w_{vs}^{vi}, w_{vi}^{vr}, w_{us}^{vs}, w_{ui}^{vi}, w_{ur}^{vr}]$.

of an unvaccinated and suspected individual to vaccinated and recovered one in one time step without even getting infected.

This assumption is also beneficial for theoretically possible transitions, where, at the same time step, not one but two or more processes change their states. First of all, it greatly simplifies defining the impact of one process on another. Furthermore, looking ahead, it greatly simplified the implementation of *Network Diffusion*. Let us consider all theoretically possible transitions $us \rightarrow vi$, (unvaccinated, suspected) $\rightarrow$ (vaccinated, infected). Let us also assume that the fact of being vaccinated significantly reduces the probability of getting infected. Then we have three ways to get from us to vi :

- 1) (unvaccinated, suspected) $\xrightarrow{w_{us}^{vs}} (vaccinated, suspected) \xrightarrow{w_{vs}^{vi}} (vaccinated, infected)$
- 2) (unvaccinated, suspected) $\xrightarrow{w_{us}^{ui}} (unvaccinated, infected) \xrightarrow{w_{ui}^{vi}} (vaccinated, infected)$
- 3) (unvaccinated, suspected) $\xrightarrow{w_{us}^{vi}} (vaccinated, infected)$

In such a context, it is easy to see that setting $w_{vs}^{vi} \ll w_{us}^{ui}$ models the real-world scenario that vaccination significantly reduces the probability of infection. The other weights are also intuitive to interpret, with one exception. The value of w_{us}^{vi} , makes impossible to deduce the relation between T_1 and T_2 .

C. Generalisation of modelling multiple processes

Using the example presented in the sec. III-B, it is possible to define a general scheme of the multiprocess model that is used in the *Network Diffusion*:

Definition 3.2: The mutual impact between n processes meeting def. 3.1, can be represented by a n -dimensional orthogonal grid (spanned by points formed by the Cartesian product of the sets of states of each process), in which the connections between points exist only in directions along the axes denoting processes (i.e. transitions are allowed only between points differing by a single coordinate).

This approach sufficiently answers the two questions posed at the beginning of this section; i.e. it allows us to define any number of processes within a single experiment and to determine their influence on each other clearly.

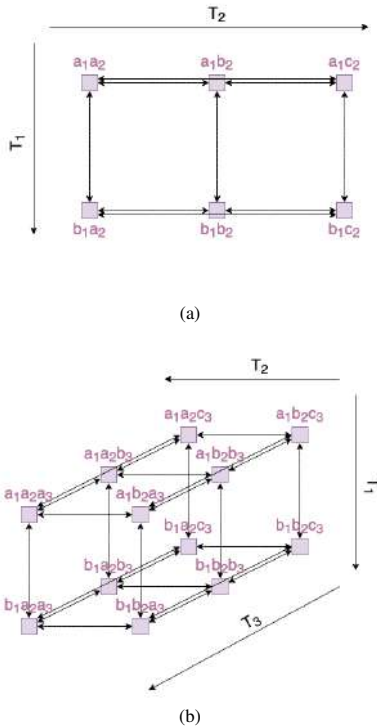


Fig. 2. General examples of multiprocess models (top) $\mathfrak{M} = (T_1 \times T_2, W)$, $T_1 = [a_1, b_1]$, $T_2 = [a_2, b_2, c_2]$; (bottom) $\mathfrak{M} = (T_1 \times T_2 \times T_3, W)$, $T_1 = [a_1, b_1]$; $T_2 = [a_2, b_2]$; $T_3 = [a_3, b_3, c_3]$; For readability of diagrams vectors of weights W are not depicted.

D. General simulation framework

The concepts presented allow us to assume that there is only one process running in a given layer of the network. This means that the network should have as many layers as many processes are simulated. As a result, with the multilayer structure, each node will be described in the context of each of the defined processes. However, if the layers do not have

an identical set of vertices, it will mean that a given node is indifferent to the process in whose layer it does not exist (e.g., a person who does not use social media and cannot be affected by information spreading via Twitter).

Continuing this line of reasoning, one can arrive at a general simulation scheme. For the convenience of the reader, it is presented as alg. 1. This allows simulations to be run on custom, discrete models with a virtually infinite number of processes and internal states.

Algorithm 1 Schema of the simulation in *Network Diffusion*.

Require: $\mathfrak{M}$

$$\mathfrak{M} = (T_1 \times \dots \times T_k, W) \quad (1)$$

Require: M

$$M = (N, L, V, E) : \quad (2)$$

$$\bigwedge_{i \in \{1, \dots, k\}} L_i \rightarrow T_i;$$

$$\bigwedge_{n \in N} n \rightarrow \tau_n = \{\tau_n^1, \dots, \tau_n^k\} \equiv \{t_1 \in T_1 \vee \emptyset, \dots, t_k \in T_k \vee \emptyset\};$$

Require: ϵ

$$\epsilon = \{1, \dots, \epsilon\} \quad (3)$$

```

1: procedure  $X(\mathfrak{M}, M, \epsilon)$ 
2:   for  $\epsilon$  in  $\epsilon$  do
3:     for  $l$  in  $L$  do
4:       for  $v$  in  $\{N \times l\}$  do
5:         for  $\bar{v}$  in  $\{(N \times l) - v\}$  do
6:           if  $\tau_v^l \neq \tau_{\bar{v}}^l$  then
7:             if  $\text{BernoulliTrial}(p = w_{\tau_n^l - \tau_v^l + \tau_{\bar{v}}^l}^{\tau_n^l}) = 1$  then
8:                $\tau_v^l := \tau_{\bar{v}}^l$ 
9:             go to line 4
10:          end if
11:        end if
12:      end for
13:    end for
14:  end for
15: end procedure

```

IV. Network Diffusion PACKAGE

Keeping in mind the way we defined the multiprocess interaction framework, it is possible to move on to the package description. We also find it useful to present some high-level aspects of its architecture.

The most important functional requirements were paired with the questions posed in the introduction of sec. III, i.e. the possibility (1) to simulate any number of processes during the experiment and (2) to define interactions between them. We also wanted to support experiments on external data (the commonly used format to store multilayer networks - *.mpx* has been chosen) and generate data and figures reflecting changes of states in the network during experiments.

When it comes to nonfunctional requirements, the main was the programming language. Due to the popularity of particular

ones in data science, we faced an alternative: Python or R. The popularity and support of both languages are very similar, so in the end, the first one was chosen subjectively. It is worth noting that we were not focused on efficiency. Therefore, the implementation was reflecting alg. 1 without optimisation.

A. Implementation

The whole package is based on the NetworkX library since, as shown in sec. II-B, it has a lot of built-in functionalities, is widely used, and has great community support. This significantly reduces the so-called entry threshold of the package for new users. Following the theoretical concepts, we divide the code into three main components: *MultilayerNetwork*, *PropagationModel*, *MultiSpreading*.

The *MultilayerNetwork* was able to store multilayer networks (as a dictionary of *networkx.Graph* objects keyed by the layer names). We also added integrated I/O methods with the NetworkX library for *.mpx* files.

As for the *PropagationModel* class, the biggest problem to solve was to define the model unambiguously according to def. 3.2. We solved it by defining the following procedure of propagation model initialisation: (1) user passes processes to the model in the form of name and internal states for each process, (2) user inserts the default weight for transition, (3) user calls a compilation function (then the model is converted to an interpretable form for the simulation environment), (4) user manually changes the weights of particular transitions (usually those which are significant for the experiment). Regarding the problem of defining a data structure to keep the model in, during compilation, a unique graph with only possible transitions has been created for each process. This made easy referring to particular transitions in the model by operating strings of type *<state>.<process>*, similar to the SimInf library II-A.

The *MultiSpreading* was compacting the modules described already into a coherent whole. It had two fields *_network* and *_model*, which were associated with each other by process names (i.e. for each process in the model, there had to be a corresponding layer in the network with the same name). The other relevant methods were *set_initial_states* and *perform_propagation*. The first one was used to set the initial conditions of the simulation - what percentage/number of nodes is currently in which state? The second method, on the other hand, was responsible for the simulation itself, according to the alg. 1. To separate the preprocessing of the experiment and the results from the simulation, a class *ExperimentLogger* has been prepared, whose object is returned after a successful experiment run.

V. EXAMPLE OF USAGE

Finally, a practical example can be presented. We prepared three comprehensive simulation scripts for this purpose: (1) example of epidemic propagation combined with vaccinations and awareness of the population, (2) marketing campaign of two competitive products, (3) gossip spreading on two different social networks. All are included in the GitHub

TABLE IV
ARTEFACTS RELATED TO THE *Network Diffusion* PACKAGE.

Artefact	Link	Description
Main repository	https://github.com/anty-filidor/network_diffusion	Main repository with package's code [1]
PyPi package	https://pypi.org/project/network-diffusion/	Built package ready to download
Anaconda package	https://anaconda.org/anty-filidor/network_diffusion	Built package ready to download
Documentation	https://network-diffusion.readthedocs.io	Web page with comprehensive documentation
Examples (source code)	https://github.com/anty-filidor/network_diffusion_examples	Auxiliary repository with examples
Runnable capsule with examples	https://codeocean.com/capsule/8807709/tree/v4	Examples ready to run online via web browser [41]

repository [1] and the Code Ocean capsule [41]. In this section, we present the first one (i.e. *epidemic.py*) in shortened form¹. To avoid copying scripts, we will refer the reader to the corresponding lines of code in the repository/capsule.

A. Scenario to simulate

In this experiment, we will simulate the spreading of three processes: an illness affecting the network, awareness of its existence, and vaccination against it. Keeping in mind the notation introduced in tab. III we will first define a propagation model:

- *illness* : $T = [s, i, r]$, $W = [w_s^i, w_i^r]$, where s denotes suspected node, i infected and r recovered,
- *awareness* : $T = [n, a]$, $W = [w_n^a]$, where n means not-aware node and a respectively aware one,
- *vaccination* : $T = [u, v]$, $W = [w_u^v]$, where u means not vaccinated node and v vaccinated one.

The compiled model can be represented by a three-dimensional grid with twelve points. Regarding the fact that we defined only a subset of all possible transitions (e.g., w_i^s was discarded), we have to define "only" twenty ones:

- for states varying in the *illness*: $w_{snu}^{inu}, w_{inu}^{rnu}, w_{sau}^{iau}, w_{iau}^{rau}, w_{snv}^{inv}, w_{inv}^{rnv}, w_{sav}^{iav}, w_{iav}^{rav}$,
- for states varying in the *awareness*: $w_{snu}^{sau}, w_{inu}^{iau}, w_{rau}^{rau}, w_{snv}^{sav}, w_{inv}^{iav}, w_{rnv}^{rav}$,
- for states varying in the *vaccination*: $w_{sau}^{sav}, w_{iau}^{iav}, w_{rau}^{rav}, w_{snu}^{snv}, w_{inu}^{inv}, w_{rnu}^{rnv}$,

Let us also add more constraints to the experiment: we want both vaccination and awareness to reduce the probability of getting sick, and awareness of the disease should make the nodes more willing to get vaccinated. We can obtain such an effect by setting some of the transitions listed above to predefined values. Moreover, we note that some of them

¹It is worth noting that the remaining two examples utilise other mechanisms of phenomena spreading. They are all in the form of the Jupyter Notebook, and we welcome readers to investigate them.

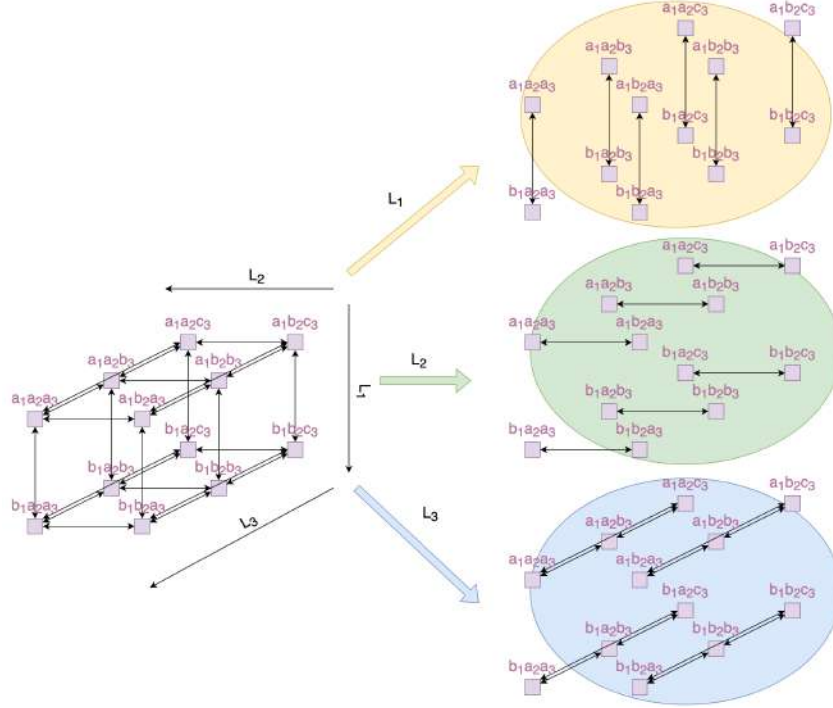


Fig. 3. Scheme for compiling a propagation model. A single, three-dimensional model is stored as a structure of three separate NetworkX graphs, one for each process, with edges covering allowed only transitions.

seem illogical. For example, w_{snu}^{snu} one, because it is hard to imagine that an individual can be vaccinated without even being aware of the disease. Hence, in our model, we can limit the transitions to the following set of non-zero ones:

- for states varying in *illness*: $w_{snu}^{inu} = 0.4$, $w_{inu}^{inu} = 0.1$, $w_{sau}^{iau} = 0.2$, $w_{iau}^{rau} = 0.3$, $w_{sav}^{iav} = 0.05$, $w_{iav}^{rav} = 0.7$,
- for states varying in *awareness*: $w_{snu}^{sau} = 0.05$, $w_{iau}^{iau} = 0.2$, $w_{snu}^{snu} = 1$,
- for states varying in *vaccination*: $w_{sau}^{sav} = 0.03$, $w_{iau}^{iav} = 0.1$.

On the other hand, in rare cases, situations like $snu \rightarrow snv$ can be possible. Therefore, we will set up some tiny weights (0.005) for all possible but unlikely to happen transitions.

B. Defining a propagation model

With these assumptions, we can finally prepare the script. As a first step, we have to import all packages (see lines 2:16). After that, we can step forward to define a propagation model (see lines 29:48). Please pay attention to the function `compile`. When the keyword argument is set, all theoretically possible transitions in the model will have a weight assigned to 0.005. With this operation, we can increase the model's flexibility to include some edge cases.

C. Defining a network

The next step is to define the network. As mentioned in sec. III-D, it should have as many layers as the processes

spreading within it. We can achieve this by duplicating a flat NetworkX graph into three layers. In this experiment, we will use a predefined graph reflecting interactions in "Les Misérables" (see lines 50:52).

D. Defining and running an experiment

After that, we can define and start the experiment (see lines 54:99). We set the initial state of the "ill" process to have 65 nodes "s", 10 "i" and 2 "r" at the first epoch. Here, we did it manually for particular nodes to obtain reproducibility, but *Network Diffusion* contains another function that does it for randomly selected nodes at once: `nx.Experiment.set_initial_states`.

E. Results and discussion

Once the simulation is completed, an `ExperimentLogger` object is returned. We can use it to prepare a report (see line 99). As a result, we obtain (snippets available in documentation - see tab. IV): propagation report for each process as set of .csv files containing information about the distributions of particular states in each epoch (*awar_propagation.csv*, *ill_propagation.csv*, *vacc_propagation.csv*), information about the network used for experiment (*network_report.txt*), the model (*model_report.txt*) and visualisation of the experiment (fig. 4a).

As we can see in fig. 4a, interactions between processes reflect the assumed scenario, the more individuals vaccinated

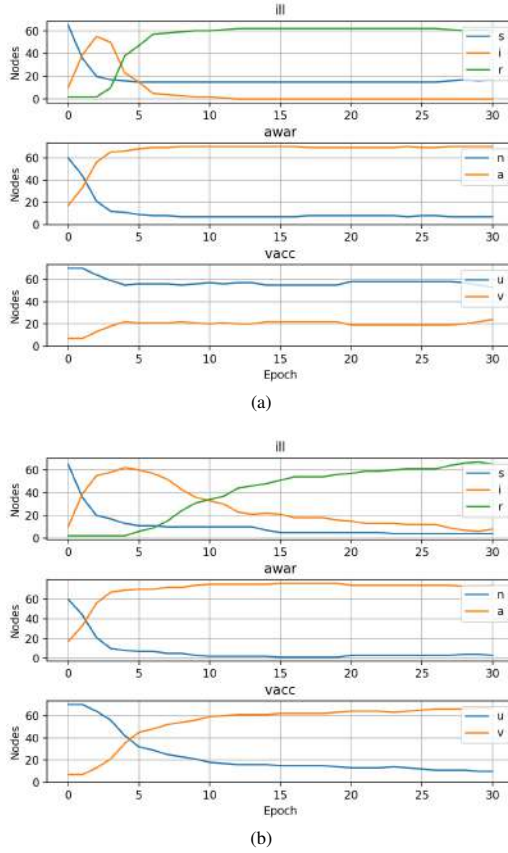


Fig. 4. Visualisations of experiments - dynamics of processes in the network (a) according to presented assumptions, (b) with modified illness to make it more contagious.

and aware, the less increase of ill nodes in the network. On the other hand, in fig. 4b, we can see a variation of the experiment. In order to obtain the effect of more malicious illnesses that cannot be so easily mitigated with vaccines and awareness of the population, we have modified weights w_{inu}^{rau} , w_{iau}^{rau} and w_{iav}^{rau} . That resulted in a slower falling curve for state i that allowed the vaccination process to continue spreading among the network.

VI. CONCLUSIONS

In this paper, we have introduced a computational approach to simulate the spreading of multiple processes in networks. Keeping in mind the analysis of the existing tools from sec. II, we find it more valuable since it fulfils the observed gap in the domain. Despite the basic character of *Network Diffusion*, we hope that it will help other researchers in analysing and understanding spreading processes in networks.

We would like to mention that there is great potential in the further development of the package. As we described in sec. III, for now, we are limited to a fairly simple schema of the experiment. We plan to extend it in the next releases

by aspects of time dynamics, such as adding layer switching cost for the spreading process [42]. We also noticed a need to provide the user with the possibility to customise a function that changes the states of nodes. Currently, we have a fairly simple mechanism that depends on iteration through neighbours combined with the Bernoulli test. However, we can use another approach, e.g. that takes into account all neighbours of the node at once. Another important aspect is an enhancement of the package's computational capabilities (see sec. A for details). Finally, the concept that is currently investigated by the authors is a development of a training mechanism that sets up weights of the models in order to fit with real data, which allows for predicting further spreading of the phenomena.

Having written all that, we can say that *Network Diffusion* may help to address the problem of multiple spreading processes within the networks. This issue is certainly real, and the COVID-19 crisis shows that we need to be able to simulate such phenomena by taking into account coexisting phenomena. As we demonstrated in the article, the package has a fine theoretical background and examples proving its usability. We strongly believe that our concept can be helpful and become an inspiration to other projects.

REFERENCES

- [1] M. Czuba and P. Bródka, "Network diffusion," 2021. [Online]. Available: https://github.com/anty-filidor/network_diffusion
- [2] P. Bródka, K. Musiał, and J. Jankowski, "Interacting spreading processes in multilayer networks: a systematic review," *IEEE Access*, vol. 8, pp. 10 316–10 341, 2020.
- [3] A.-L. Barabási and M. Pósfai, *Network Science*. Cambridge: Cambridge University Press, 2016, ch. 10.1, pp. –.
- [4] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, "Biogrid: A general repository for interaction datasets," *Nucleic acids research*, vol. 34, pp. D535–D539, 2006.
- [5] M. De Domenico, A. Solé-Ribalta, S. Gómez, and A. Arenas, "Navigability of interconnected networks under random failures," *National Academy of Sciences*, vol. 111, no. 23, p. 8351, 2014.
- [6] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [7] P. Kazienko, K. Musiał, E. Kukla, T. Kajdanowicz, and P. Bródka, "Multidimensional social network: Model and analysis," in *International Conference on Computational Collective Intelligence*. Springer, 2011, pp. 378–387.
- [8] P. Bródka, A. Chmiel, M. Magnani, and G. Ragozini, "Quantifying layer similarity in multiplex networks: A systematic study," *Royal Society open science*, vol. 5, no. 8, p. 171747, 2018.
- [9] M. Kivela, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of complex networks*, vol. 2, no. 3, pp. 203–271, 2014.
- [10] M. Magnani and L. Rossi, "The ml-model for multi-layer social networks," in *2011 International Conference on Advances in Social Networks Analysis and Mining*. IEEE, 2011, pp. 5–12.
- [11] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM Sigmod Record*, vol. 42, no. 2, pp. 17–28, 2013.
- [12] W. O. Kermack and A. G. McKendrick, "A contribution to the mathematical theory of epidemics," *royal society of london. Series A, Containing papers of a mathematical and physical character*, vol. 115, no. 772, pp. 700–721, 1927.
- [13] R. A. Holley and T. M. Liggett, "Ergodic theorems for weakly interacting infinite systems and the voter model," *The annals of probability*, pp. 643–663, 1975.
- [14] F. M. Bass, "A new product growth for model consumer durables," *Management science*, vol. 15, no. 5, pp. 215–227, 1969.
- [15] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *9th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, p. 137.

- [16] H. Zang, "The effects of global awareness on the spreading of epidemics in multiplex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 492, pp. 1495–1506, 2018.
- [17] M. Scatà, A. Di Stefano, P. Liò, and A. La Corte, "The impact of heterogeneity and awareness in modeling epidemic spreading on multiplex networks," *Scientific reports*, vol. 6, no. 1, 2016.
- [18] P. Shakaran, A. Bhatnagar, A. Aleali, E. Shaabani, and R. Guo, *Diffusion in Social Networks*. Springer: Springer, 2016, ch. 4.
- [19] W. V. d. Broeck, C. Gioannini, B. Gonçalves, M. Quagiotto, V. Colizza, and A. Vespignani, "The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale," *BMC Infectious Diseases*, vol. 11, no. 1, p. 37, Feb 2011. [Online]. Available: <https://doi.org/10.1186/1471-2334-11-37>
- [20] G. Rossetti, L. Milli, S. Rinzivillo, A. Sirbu, D. Pedreschi, and F. Gnanotti, "Ndlb: A python library to model and analyze diffusion processes over complex networks," *Int. J. Data Sci. Anal.*, vol. 5, no. 1, pp. 61–79, Feb 2018.
- [21] S. Widgren, P. Bauer, R. Eriksson, and S. Engblom, "SimInf: An R package for data-driven stochastic disease spread simulations," *Journal of Statistical Software*, vol. 91, no. 12, pp. 1–42, 2019.
- [22] F. Alvarez, P. Crépey, M. Barthélemy, and A.-J. Valleron, "Sisspread: A software to simulate infectious diseases spreading on contact networks," *Methods of information in medicine*, vol. 46, p. 19, 2007.
- [23] J. Douglas, S. Bianco, S. Edlund, T. Engelhardt, M. Filter, T. Günther, M. HuKun, E. Nixon, N. Sevilla, A. Swaid, and J. Kaufman, "Stem: An open source tool for disease modeling," *Health security*, sie 2019.
- [24] Franz Inc., "Allegrograph website," [26 May 2022]. [Online]. Available: <https://allegrograph.com/products/allegrograph>
- [25] E. Inc., "Arcgis pro - 2d and 3d gis mapping software," [26 May 2022]. [Online]. Available: <https://www.esri.com/en-us/arcgis/products/arcgis-pro/overview>
- [26] Cytoscape Consortium, "Cytoscape 3.8.2 user manual," [26 May 2022]. [Online]. Available: <https://manual.cytoscape.org/en/stable>
- [27] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," *International AAAI Conference on Weblogs and Social Media*, 2009.
- [28] G. Malewicz, M. H. Austern, A. J. Bik, J. C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski, "Pregel: A system for large-scale graph processing," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 135–146. [Online]. Available: <https://doi.org/10.1145/1807167.1807184>
- [29] J. Ellson, E. R. Gansner, E. Koutsofios, S. C. North, and G. Woodhull, *Graphviz and Dynagraph — Static and Dynamic Graph Drawing Tools*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, ch. -, pp. 127–148.
- [30] G. Csardi and T. Nepusz, "The igraph Software Package for Complex Network Research," *InterJournal*, vol. Complex Systems, 2006. [Online]. Available: <https://igraph.org>
- [31] M. Matteo, V. Davide, and D. Mikael, *Multinet: Analysis and Mining of Multilayer Social Networks*, 2020. [Online]. Available: <https://cran.r-project.org/web/packages/multinet/index.html>
- [32] M. De Domenico, M. A. Porter, and A. Arenas, "Muxviz: A tool for multilayer analysis and visualization of networks," *Journal of Complex Networks*, vol. 3, no. 2, pp. 159–176, 10 2014.
- [33] Neo4j, Inc., "Neo4j graph platform the leader in graph databases," [26 May 2022]. [Online]. Available: <https://neo4j.com>
- [34] G.-H. Ghim, N. Cho, and J. Seo, *NetMiner*. New York, NY: Springer New York, 2014, pp. 1025–1037.
- [35] C. T. Butts, *Network: Classes for Relational Data*, The Statnet Project (<http://www.statnet.org>), 2020, r package version 1.16.1. [Online]. Available: <https://CRAN.R-project.org/package=network>
- [36] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using networkx," in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., Pasadena, CA USA, 2008, pp. 11 – 15.
- [37] M. Smith, A. Ceni, N. Milic-Frayling, B. Shneiderman, E. M. Rodrigues, J. Leskovec, and C. Dunne, *NodeXL: A Free and Open Network Overview, Discovery and Exploration Add-in for Excel 2007/2010/2013/2016*, 2010. [Online]. Available: <http://nodexl.codeplex.com/>
- [39] D. Auber, D. Archambault, R. Bourqui, M. Delest, J. Dubois, A. Lambert, P. Mary, M. Mathiaut, G. Melancon, B. Pinaud, B. Renoust, Mikko Kivelä, "Multilayer networks library for python (pymnet) documentation," Aug 2017, [26 May 2022], <http://www.mkivela.com/pymnet>.

- and J. Vallet, "TULIP 5," in *Encyclopedia of Social Network Analysis and Mining*, R. Alhajj and J. Rokne, Eds. Springer, Aug. 2017, pp. 1–28. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01654518>
- [40] yWorks GmbH, "yEd graph editor," [26 May 2022]. [Online]. Available: <https://www.yworks.com/products/yed>
- [41] M. Czuba and P. Bródka, "Demo-code for the paper "network diffusion: A package to simulate spreading of multiple interacting processes in complex networks","" 11 2021. [Online]. Available: <https://codeocean.com/capsule/8807709/tree/v3>
- [42] B. Min, S.-H. Gwak, N. Lee, and K.-I. Goh, "Layer-switching cost and optimality in information spreading on multiplex networks," *Scientific reports*, vol. 6, no. 1, pp. 1–12, 2016.

APPENDIX

A. Computational efficiency of the package

Since *Network Diffusion* has been considered a prototype framework, we have decided to investigate its time efficiency compared to NDLib [20]. We performed an experiment on a single-layer SIR model (due to limitations of the NDLib, we could not test a more complex case) spreading within the Erdos-Renyi graph, the size of which varied. For each network size, both implementations were executed 10 times (to mitigate the side effects that burden the processor). Each call to the function was measured in milliseconds. The results are presented in fig. 5. The time efficiency of Network Diffusion is a field for improvement, especially for large networks.

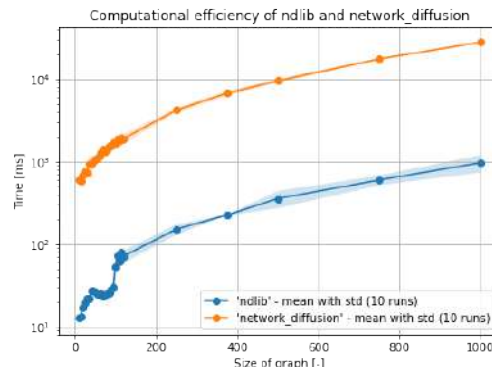


Fig. 5. Comparison of the efficiency of the Network Diffusion and the NDLib.

B. Running attached code examples

There are two ways to run the examples: (1) local execution, (2) via the Code Ocean capsule [41]. Despite the differences in code handling, both are expected to produce the same results. To follow the first, the user needs to have Python installed. It is recommended to create a virtual environment based on the *requirements.txt* file. Then, to run the chosen example, the corresponding script needs to be executed. The second method requires far less effort. To run our code, we recommend first that you duplicate the capsule into a self-owned account. Then, to run the engine, the "reproducible run" button has to be pressed. To run a Jupyter notebook, the user has to select a Jupyter icon from the list below the mentioned button.

4.2 Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks

Authors: Michał Czuba, Mateusz Nurek, Damian Serwata, Yu-Xuan Qiu, Mingshan Jia, Katarzyna Musiał, Radosław Michalski, Piotr Bródka

Published at: Big Data Mining and Analytics; August 2024

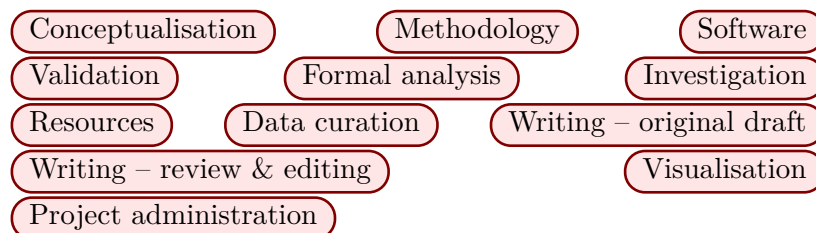
MNiSW points: 20 (IF 6.2)

Citations: Web of Science: 3

(as of 6th March 2026) Scopus: 4

Google Scholar: 5

CRedit:



BIG DATA MINING AND ANALYTICS
 ISSN 2096-0654 06/25 pp637–654
 DOI: 10.26599/BDMA.2024.9020010
 Volume 7, Number 3, September 2024

Network Diffusion Framework to Simulate Spreading Processes in Complex Networks

Michał Czuba*, Mateusz Nurek, Damian Serwata, Yu-Xuan Qiu, Mingshan Jia, Katarzyna Musiał, Radosław Michalski, and Piotr Bródka

Abstract: With the advancement of computational network science, its research scope has significantly expanded beyond static graphs to encompass more complex structures. The introduction of streaming, temporal, multilayer, and hypernetwork approaches has brought new possibilities and imposed additional requirements. For instance, by utilising these advancements, one can model structures such as social networks in a much more refined manner, which is particularly relevant in simulations of the spreading processes. Unfortunately, the pace of advancement is often too rapid for existing computational packages to keep up with the functionality updates. This results in a significant proliferation of tools used by researchers and, consequently, a lack of a universally accepted technological stack that would standardise experimental methods (as seen, e.g., in machine learning). This article addresses that issue by presenting an extended version of the Network Diffusion library. First, a survey of the existing approaches and toolkits for simulating spreading phenomena is shown, and then, an overview of the framework functionalities. Finally, we report four case studies conducted with the package to demonstrate its usefulness: the impact of sanitary measures on the spread of COVID-19, the comparison of information diffusion on two temporal network models, and the effectiveness of seed selection methods in the task of influence maximisation in multilayer networks. We conclude the paper with a critical assessment of the library and the outline of still awaiting challenges to standardise research environments in computational network science.

Key words: computational framework; seed selection; influence maximisation; spreading models; temporal networks; multilayer networks; network science; network control

1 Introduction

With the rapid growth of data generated by humankind in recent decades, the issue of their efficient processing becomes one of the main challenges that need to be addressed to boost experimental sciences. That

especially concerns irregular structures, like graphs, that are much harder to enclose in robust frameworks, such as images, audio, or text. Computational network science^[1], the primary discipline focusing on applications of graphs, has developed many tools that

- Michał Czuba, Mateusz Nurek, Damian Serwata, Radosław Michalski, and Piotr Bródka are with Department of Artificial Intelligence, Wrocław University of Science and Technology, Wrocław 50-370, Poland. E-mail: michal.czuba@pwr.edu.pl; mateusz.nurek@pwr.edu.pl; damian.serwata@pwr.edu.pl; radoslaw.michalski@pwr.edu.pl; piotr.brodka@pwr.edu.pl.
- Yu-Xuan Qiu, Mingshan Jia, and Katarzyna Musiał are with Data Science Institute, University of Technology Sydney, Ultimo, NSW 2007, Australia. E-mail: yuxuan.qiu@uts.edu.au; mingshan.jia@uts.edu.au; katarzyna.musial-gabrys@uts.edu.au.

* To whom correspondence should be addressed.

Manuscript received: 2023-10-01; revised: 2024-01-04; accepted: 2024-02-20

facilitate research-oriented computations. They are used in many tasks, like epidemiology^[2], social network analysis^[3], recommendation systems^[4, 5] or urban planning^[6], to name just a few. However, the variety and versatility of those solutions significantly stand out from frameworks designed to deal with other data modalities.

As stands in Ref. [7], the number of spreading models and problems they tackle is overwhelming and exceeds the abovementioned examples. One can add to that various network models and forms of simultaneous coexistence of many processes^[8]. When these phenomena are up to be analysed, researchers must first design and perform adequate simulations. Unfortunately, as demonstrated in our previous work^[9], there is a big proliferation of environments that can be used for that. Hence, outcomes of such research are often presented only as articles without proper code attached, which could allow reproducing results and easily include proposed methods (e.g., a new centrality measure, spreading model, etc.) in following research conducted by another organisation. Although challenges related to the standardisation of experimental environments and results reproducibility have been addressed a long time ago in areas of computer science, like computer vision, signal processing, or machine learning, they are still valid in network science.

This paper presents an enhanced version of Network Diffusion, a framework originally designed to simulate the coexisting spreading processes in multilayer networks^[9]. Since publication, it has been transformed into a comprehensive library that allows the processing of both temporal and multilayer networks with various

spreading processes on top of them. Though the tool has been developed during various research activities of the authors, we collect developed methods and models (most of them are referenced in Table 1) into one consistent entity. To the best of our knowledge, there is no such library providing similar functionalities. Therefore, this framework can be considered as complementary to other available solutions. We can summarize a contribution brought with this paper as an introduction of a research environment for simulating spreading processes in complex networks to promote open science and research reproducibility. Moreover, with the framework, we tackle and solve (in a minimal way) still valid problems from a domain of influence maximisation, epidemiology, and network modelling.

The outline of the article is the following. First, we present a review of existing solutions tackling given problems (Section 2), then the general outline of library functionalities (Section 3). In Section 4, we show four use cases of experiments conducted with the package. After that, in Section 5, we describe a study of the computational efficiency of Network Diffusion. The paper is summarized in Section 6. For the reader's convenience, we attach a table with abbreviations used in the article (Table 1).

2 Existing Tools

In this section, we would like to briefly introduce existing tools and software packages with functionalities similar to the Network Diffusion package. The detailed list and comparison are already available^[9]. Here, we would like to focus on tools that can be used for research on various spreading processes

Table 1 Abbreviations used in the paper.

Abbreviation	Abbreviation expansion	Note
SIS	Suspected-Infected-Suspected	A basic epidemiological spreading model ^[10]
SIR	Suspected-Infected-Removed	A basic epidemiological spreading model ^[10]
LTM	Linear Threshold Model	A spreading model in the sense as in Refs. [11, 12] with extension to multilayer networks as in Ref. [13]
ICM	Independent Cascade Model	A spreading model in the sense as in Ref. [12], with extension to multilayer networks similarly to Ref. [13]
SIR-UA	Suspected-Infected Removed and Unaware-Aware	Two coexisting spreading models in the sense as in Refs. [8, 9]
NEM	Network Epistemology Model	A spreading model in the sense as in Ref. [14], with extension to temporal networks as in Ref. [15]
MDS	Minimal Dominating Set	A set of network's nodes allowing to control its entire structure according to Refs. [16, 17]
CogSNet	Cognition-driven Social Network	A temporal network model according to Ref. [18]

in multilayer and temporal networks.

2.1 GLEaMviz

The first application with functionalities corresponding to the designed software is a GLEaMviz^[19]. It works with real data, population density, and migration around the world, combined with stochastic models of disease propagation. As a result, it provides a sophisticated simulation environment. Due to the large scale of the experiments (the whole world), a single node is a population of a given size (defined by the user). A very interesting feature is the manual definition of the epidemiological model. GLEaMviz makes this possible by manipulating the compartments. Allowable transitions between them are also fully definable. Users are also able to choose the geographical origin of the disease, specify the initial proportions of individuals in each compartment, determine its duration, and more. There is also an option to generate various visualisations at the end of the experiment. Despite the interesting functionalities mentioned above, GLEaMviz has a few disadvantages: it is limited to disease spreading, only allows the propagation of one process at a time, and does not support multilayer and temporal networks.

2.2 NDlib

Network Diffusion library (NDlib)^[20] is a Python package based on the NetworkX library. It allows performing simulations with many predefined epidemiological (such as SIS, SIR, etc.), influence spread (LTM, ICM, Profile, etc.), or opinion formation (Voter, Sznajd, etc.) models, and even dynamics (models with the capacity to change the topology of network). Moreover, the user can create its own customised models. Results visualisation is also possible via Matplotlib or Bokeh with the flexibility to append a custom graphical engine. NDlib also has some interesting run-time features. First, it includes an option to perform a “multi-execution” of the simulation by parallel computing. As this kind of experiment is generally stochastic, this feature gives a chance to see the general behaviour of the observed phenomena. It also enables running the simulation on a server (as well as locally). For users being unfamiliar with Python, NDlib’s authors created NDQL, a query language (based on SQL syntax) that supports elementary commands of the library. For those who cannot programme, they also provided a “visualisation

framework” to play with some of the models implemented with a graphical user interface-based tool. NDlib is a very useful library with many features. However, it does not directly support experiments where multiple spreading processes interact with each other, as well as experiments on temporal networks.

2.3 SimInf

The next tool is SimInf — a framework for data-driven stochastic disease spread simulations^[21], a package for R language. This tool allows a user to define custom spreading models that are executed following a principle of continuous-time Markov chains and the Gillespie stochastic simulation algorithm. Interestingly, no network is needed to run the simulation, as the algorithms work on the assumption of homogeneous mixing and additional data, such as births, deaths, and population movements at predefined time points. After a successful simulation, it is possible to display a summary graph. Nevertheless, we must note that no support for operations on real, multilayer, and temporal networks, as well as the lack of an option to define interacting processes, are its distinct shortcomings.

2.4 Sispread

Sispread^[22] is a simple application implemented in C language. It is a console tool without a graphical interface. It focuses on epidemic models (SI, SIS, SIR) with the possibility of deep analysis of the performed experiments. This tool supports very basic IO operations — the user can upload a custom network that meets certain requirements (without the option to manipulate it), or generate it using one of three available models: Barabási-Albert, Erdős-Renyi, or Kleinberg. As a result of the experiment, numerical data are returned for further analysis. Similarly to previous packages, it does not support multiple spreading processes, as well as multilayer or temporal networks.

2.5 STEM

Another tool is Spatio-Temporal Epidemiological Modeler (STEM)^[23]. As an extension of the Eclipse development environment, it uses its graphical layout and the general philosophy of user interaction. It requires the user to specify a medium where the simulation is performed with its discretisation level (i.e., whether the node is a municipality or an entire county). The next step is to attach an appropriate solver

propagation model and set the starting parameters of the experiment. Once this is done, it is possible to visualize the spread progress. In addition to the extensive user interface, there is an option to run simulations in headless mode, accessible from the terminal. This software also allows for simulating (to a limited extent) the propagation of coexisting phenomena, such as vaccine vs. disease. However, this package is primarily designed for epidemiologists, and all of its functionalities are related to this subject. Therefore, like other tools, it does not support spreading in multilayer or temporal networks.

2.6 EpiModel

EpiModel^[24] is an R tool for simulating infectious disease dynamics. It contains deterministic compartmental models, stochastic individual-contact models, and stochastic network models. Network models use the robust statistical methods of Exponential-family Random Graph Models (ERGMs). As a standard, it includes SI, SIS, and SIR models. EpiModel does not directly support real networks (it focuses on generated ones) and multiple processes spreading at the same time. It also focuses on epidemic models. Unlike previously described tools, thanks to the application of ERGMs, it partially supports multilayer and temporal networks.

2.7 Summary

Based on our review of existing tools, we can conclude that, to the best of our knowledge, none of them supports multiple processes at the same time (like information and disinformation, virus and information about the virus, or multiple diseases), and most of them do not support multilayer and temporal networks. That is why we focus on these functionalities in Section 4, where we present a few examples of experiments using the Network Diffusion library.

3 Feature Overview

After a thorough review of the available conclusions from Section 2 are a principal motivation for investing time and human resources in developing the Network Diffusion package. In formulating its design, we operate under the following four premises:

- (1) Compatibility with other tools commonly used in the domain of data science;
- (2) Development of a tool in the form of a

framework as opposed to a library comprising loosely connected code fragments;

- (3) Supporting both multilayer and temporal networks; and

- (4) Supporting spreading models with discrete states.

By releasing a Python-based tool, we naturally facilitate access to the entire ecosystem associated with that language. Primarily, we maintain compatibility with the widely-used graph processing package, NetworkX, which is the backbone of Network Diffusion. It is good to note, that some parts of the library are implemented in C language — that allows to speed up computations with high computational cost.

The framework-oriented design approach enables the straightforward extension of the tool with new spreading models. That is important, especially in research on social influence, misinformation, or viral marketing, where nuanced phenomena require modelling tailored to their unique nature. Furthermore, this approach allows for incremental project advancement (e.g., as a side effect of in-depth research).

As outlined in Section 2, only a few tools for simulating spreading phenomena operate on temporal or multilayer networks, despite these graph models offering a more accurate reflection of reality. This fundamental disparity leads us to present our internal research environment as a packaged solution. It is good to note that during the implementation of network models, the library is enriched with centrality methods tailored to these specific structures. These measures can be efficiently used to select seed nodes for diffusion processes.

The assumption regarding the discrete nature of dissemination models undoubtedly constitutes a tool limitation. However, the authors have yet to encounter the need to use a different class of models in their research so far. Therefore, this aspect, being unobjectionable, has been omitted.

While describing the functionality, it is worth mentioning the experiment scheme facilitated by this library. The process entails three fundamental steps: defining the propagation model, specifying the network upon which the model is executed, and determining the simulation parameters (e.g., number of steps). With these three components in place, experimenting becomes feasible, yielding precise data concerning the network's state at each simulation step. Section 4.1

demonstrates this procedure using the appropriate modules within the package, employing a selected research problem.

4 Experiments with Network Diffusion

This section presents four challenges from the domain of spreading models, which are tackled using the Network Diffusion library. Please note that all experiments described below are intended to meet two goals. Firstly, various functionalities of the Network Diffusion library are presented. Secondly, despite their compactness, they are big enough to draw particular conclusions regarding issues like phenomena spreading or network modelling. Regarding the above, we do not provide a comparative analysis of the methods used — this paper is not a place for that, and dwelling on such details could disturb the article's main message.

All experiments are conducted in an environment based on Python 3.10 with Network Diffusion 0.13.0. Though simulations are executed parallelly, we use three computers for that. The first one, which serves in problems from Sections 4.1 and 4.3, has been installed macOS 13 with M2 Pro CPU (arm64 architecture) and 32 GB RAM. The second one is used in experiments from Section 4.2 and has been installed Ubuntu 20.04 with Intel Core i7-6500U (x86_64 architecture) CPU and 16 GB RAM. Simulations presented in Section 4.4 are conducted on the third one on a workstation with Windows 10 Pro (22H2), whose hardware consists of Intel Core i7-6820HQ (x86_64 architecture) CPU and 32 GB RAM. It is good to note that we provide the source code enabling the reproduction of all experiments. It can be found in the repository available on GitHub (<https://github.com/anty-filidor/bdma-experiments>) with the instructions on how to run it.

4.1 Problem: Simultaneous spreading of disease and awareness of the threat

In this Section, we demonstrate how to define a custom spreading model using pre-existing library interfaces and how to conduct an experiment. Two processes (disease and awareness of its existence) interacting with each other will serve as examples. First, the SIR epidemiological model with parameters specific to COVID-19^[25], namely the infection rate coefficient (α) and recovery rate coefficient (β). According to the simulated scenario, the second phenomenon, i.e., awareness of the disease Unaware-Aware (UA), is

supposed to reduce the infection rate among agents who are conscious of the contagion. Finally, three cases are considered: minimising social interactions (e.g., lockdowns), wearing masks, and the absence of infection countermeasures. With the experiment, we will determine the effectiveness of the two measures mentioned above in combating the rise of the epidemic.

4.1.1 Problem formulation

That scenario can be depicted as a state graph with transitions between states, as shown in Fig. 1. The horizontal transitions occur within the disease context, while the vertical transitions occur within the awareness. For instance, the transition $SA \rightarrow IA$ signifies that an agent's state change from suspected and aware to infected and aware takes place with a probability of α' .

It is important to note that the recovery rate does not depend on the awareness of the disease, i.e., $IU \rightarrow RU$ and $IA \rightarrow RA$ both have the same weight denoted as β . On the other hand, concerning the infection rate, we have assumed that α' will be a reduced value of α by a factor λ appropriate to the simulated scenario (see Table 2). Thus, for lockdown, we set infection risk reduction to 90% ($\lambda = 0.1$)^[26], for wearing masks or 1m social distancing to 65% ($\lambda = 0.35$)^[27, 28], and for no countermeasures applied no risk reduction will take place, hence $\lambda = 1$.

In the case of the UA process, we assume that agents initially become aware of the disease with a constant and low probability γ . However, if they get infected,

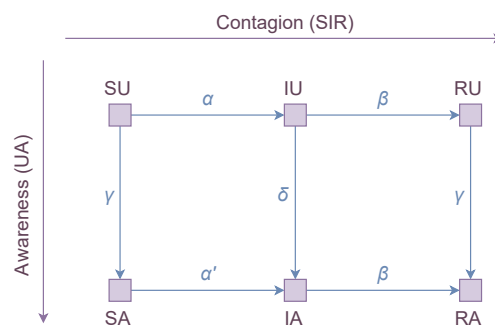


Fig. 1 Graph of states and transitions between them for the spreading model of two processes: contagion (SIR) and awareness of its existence (UA). Each state is represented by two letters indicating the state of both processes, e.g., IU indicates that the node is Infected (I) with the disease and Unaware (U) of its existence. The symbols on the arrows indicates the transition probability from one state to another (for values please see Table 2).

Table 2 Transition probabilities (or weights) and their interpretations in the SIR-UA model.

Symbol	Formula/Value	Description
α	0.19	Probability of infection for unaware agents
α'	$\lambda\alpha$; $\lambda \in \{0.1, 0.35, 1\}$	Probability of infection for aware agents
β	0.10	Probability of recovery
γ	0.01	Probability of awareness for uninfected agents
δ	$\gamma + 1 - 0.3$	Probability of awareness for infected agents

this value increases proportionally to the presence of symptomatic cases in the population observed during the COVID-19 pandemic (70% of people having COVID-19 have symptoms of the disease)^[29]. Transition weights between states are also included in Table 2.

Another crucial aspect is the environment where the processes propagate. In this case, we assume propagation occurs in a multilayer network with two layers. In one layer (e.g., a communication network), awareness of the disease spreads, while in the second layer (e.g., a network of physical contacts), the disease itself. Agents can change their states based on interactions with their neighbours, with the probability of adopting the neighbour's state corresponding to the transition weight according to Fig. 1 and Table 2. An exception is the transitions $IU \rightarrow RU$, $IA \rightarrow RA$, for which no interaction with neighbouring agents is needed for the transitions to occur.

4.1.2 Experiment setup

In the repository attached to the article, within the script `utils/models.py`, we have included the source code for the `SIR_UAModel` class, which represents the implementation of the problem discussed in this section. It also demonstrates how to extend the Network Diffusion library with custom spreading processes. To create a specific model, one needs to extend the base class (which is `network_diffusion.models.BaseModel`) and concrete the six methods accordingly.

The constructor (`SIR_UAModel.__init__`) accepts nine parameters, of which six are the transition weights, and the remaining two are the initial percentage of infected agents and aware agents. The class utilises the transition graph introduced in our previous work^[9], which helps in determining the potential state change of the evaluated agent during the simulation.

The following method determines the initial state of

the network before the start of the simulation (`SIR_UAModel.determine_initial_states`). In this function, seeds are chosen, that is, the infected agents and aware agents from which the process will propagate. The method returns a list of `NetworkUpdateBuffer` structures, which contain the necessary information for updating the network state (specifically, the agent's ID and states in each layer). In this problem, we aim to determine the impact of λ on the number of infections. Therefore, seed nodes are decided to be chosen randomly — that approach is the most transparent in presented circumstances. It is good to add that we will select initially infected and aware actors independently, with budgets equal to 5%.

The function `SIR_UAModel.agent_evaluation_step` is called during each simulation step for each agent individually. It takes the agent's ID, the ID of the layer for which the evaluation takes place, and the network itself. This function determines the state the agent will adopt in the next step of the experiment. That is done by reading the set of possible states the agent can fall in, and then iteratively attempting to adopt the neighbour's state with the probability as depicted in Fig. 1. The output of this function is the new state of an agent in the given network layer.

A function `SIR_UAModel.network_evaluation_step` is responsible for evaluating the entire network in a single experiment step. In this case, it involves iterating over each layer and each agent within that layer to determine its new state according to the method (`SIR_UAModel.agent_evaluation_step`). The output is a list of `NetworkUpdateBuffer` structures.

The last two methods are needed to generate the experiment report while using this model. These are `_str_`, which provides a description of the model, and `get_allowed_states`, which returns the textual form of the graph presented in Fig. 1.

A model prepared this way can be executed on a given two-layer network with the `network_`

`diffusion.Simulator` class^{*}. By using the `Simulator.perform_propagation` method, we conduct a simulation that conceptually consists of eight steps[†] — see Algorithm 1.

4.1.3 Results & discussion

As part of the experiment, we utilize three types of networks: AUCS (a real graph obtained from interactions between employees of Department of Computer Science, Aarhus University, Denmark), Scale-Free, and Erdős-Rényi. For each graph, the simulation is repeated 50 times, and the obtained results are averaged. Figures 2–4 depict the results in the form of infection and awareness curves within the population for each evaluated λ value. It can be

Algorithm 1 Simplified simulation procedure in Network Diffusion library

```

1: procedure perform_propagation(network, model, epochs)
2:   states_0 ← model.determine_initial_states();
3:   model.update_network(states_0)  ▷Predefined
   function in the BaseModel class;
4:   for e in [1, 2, ..., epochs] do
5:     states_e ← model.network_evaluation_step
       (network);
6:     model.update_network(network, states_e);
7:   end for
8:   logs ← generate logs from experiment;
9:   return logs;
10: end procedure

```

observed that limiting the infection rate makes sense for every network type. Another conclusion is that the network structure clearly influences the course of the processes. The highest percentage of infected agents at a given time is observed in the random network, accompanied by a rapid saturation of the UA process. We can observe that the propagation dynamics are significantly lower for the Scale-Free graph and AUCS network, where degree distribution is not as uniform as for the Erdős-Rényi model. Consequently, the number of infected agents is lower as well.

4.2 Problem: Linear threshold model in temporal networks

4.2.1 Problem formulation

The models of social influence consider different scenarios of the spread. In a binary-LTM that has been introduced in Ref. [11], the adoption of a belief occurs in a situation where a certain fraction of the neighbours in an ego network of a node shares a new belief. In this case, contrary to the stochastic-ICM, we can consider the model deterministic and — given that only unidirectional change is possible — the only stop condition of the model is when no future activations of nodes are possible.

Typically, the LTM model is used in a static network scenario, where the set of nodes and edges is fixed. However, given that this approach is simplistic compared to real-world networks^[30], it is worth

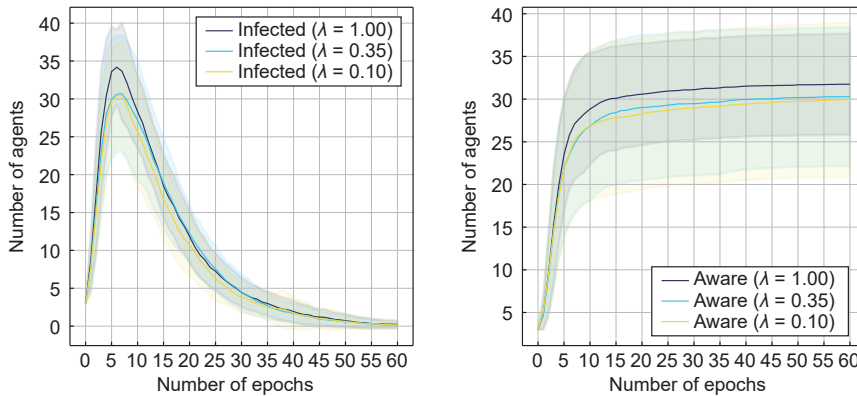


Fig. 2 Infection and awareness curves for spreading of SIR-UA model within aucs-2 network in three different epidemic regimes, where we can observe how different prevention measures lockdown (infection risk reduction by 90% — $\lambda = 0.10$), wearing masks or 1 m social distancing (risk reduction by 65% — $\lambda = 0.35$), and no measures (no risk reduction — $\lambda = 1.00$) affect the number of infected and aware individuals (agents) in the network.

^{*} In the initial version of the package, it is called `network_diffusion.MultiSpreading`.

[†] Simulation algorithm is described in a simplified form, and simulations in temporal networks are not covered by the pseudocode.

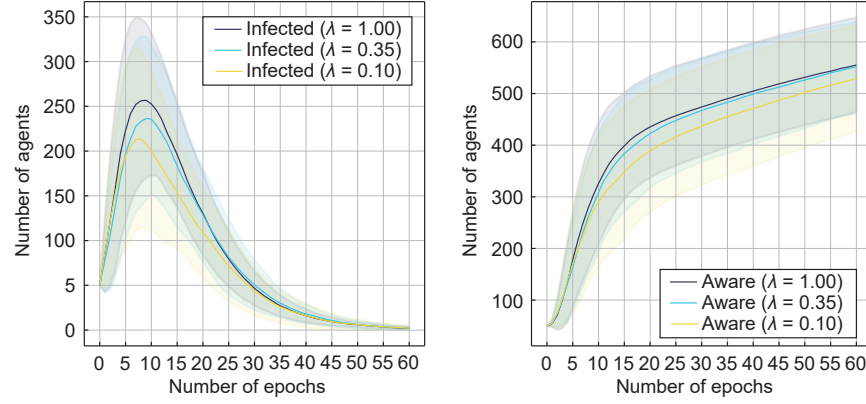


Fig. 3 Infection and awareness curves for spreading of SIR-UA model within sf-2 network in three different epidemic regimes, where we can observe how different prevention measures lockdown ($\lambda = 0.1$), wearing masks or 1 m social distancing ($\lambda = 0.35$), and no measures ($\lambda = 1$) affect the number of infected and aware individuals (agents) in the network.

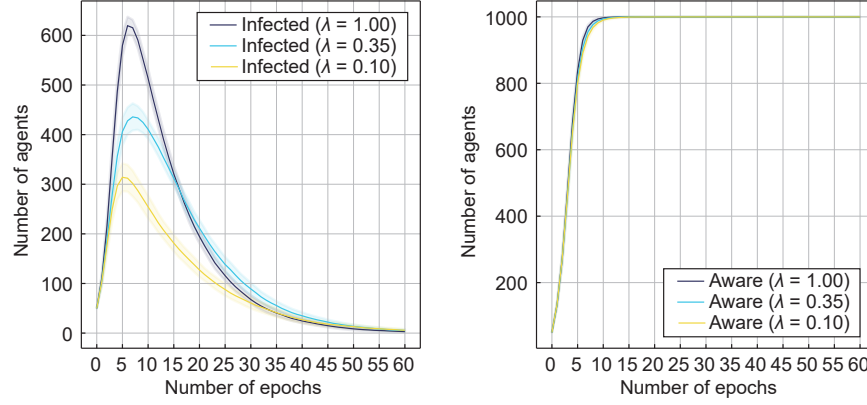


Fig. 4 Infection and awareness curves for spreading of SIR-UA model within er-2 network in three different epidemic regimes, where we can observe how different prevention measures lockdown ($\lambda = 0.1$), wearing masks or 1 m social distancing ($\lambda = 0.35$), and no measures ($\lambda = 1$) affect the number of infected and aware individuals (agents) in the network.

considering the relaxation of the time constraint and moving the problem of seed selection to time-varying networks^[31, 32]. Yet, the change of the setting to temporal networks also requires an adequate choice of a temporal network model, which can be considered as a problem by itself. As we already mentioned, for the static scenario the stop condition is when no further activation of nodes is possible. Yet it is worth underlining that for the dynamic setting the activations can still be possible due to the network dynamics, so here the process can evolve for longer.

The goal of this experiment is to answer the question of how the network dynamics change the final spread of the LTM compared to the static scenario. Conceptually one can think of such an experiment in a way that the answer to this question will allow for

understanding whether there is a speed-up or slow-down of diffusion dynamics when the network evolves over time^[33].

4.2.2 Experiment setup

In the experiment on the spread of influence following the LTM model, the CogSNet model will be used^[18]. The properties of this model — contrary to snapshot-based approaches — allow for continuous modelling of temporal networks. Here, each event that happens at a time point either creates an edge between nodes interacting with each other or amplifies it. Next, as time passes, the weights of the edges decay over time and in case of no forthcoming events lead to the vanishing of edges. Otherwise, if events occur, they again lead to an increase in weight. The decaying function is typically a power or exponential function.

This approach, described in more detail in Ref. [18] allows for fine-grained modelling contrary to incremental or snapshot-based temporal network models. However, as there is no continuous variant of the LTM model (the closest to it being introduced in Ref. [34]), to match the temporal network with the LTM, we will be matching the iterations of the spreading model with intervals of time in the CogSNet network. The stop condition for the LTM is when no further activations are possible and — for the temporal case — the process is run for no later than the last edge appeared.

To investigate how much the temporal network speeds up or slows down the spread of influence, the experimental setting is as follows. For experiments, we use the dataset on email exchange in manufacturing company^[35]. For the LTM, we focus on evaluating the seeding budget γ and the threshold μ for node activations. For the seed selection strategy, a random seed selection has been applied, thus the experiments are repeated 100 times. As for the CogSNet model parameters, based on the work^[18], we use the exponential forgetting function, edge lifetime of seven days, edge removal threshold $\theta = 0.1$, and forgetting function parameter $\lambda = 0.8$, with the units of hours. This set of parameters of the model in the aforementioned work results in an adequate match of human memory imprints and ground truth data. The parameters of the experiments alongside their explanation are shown in Table 3.

4.2.3 Results & discussion

The results of the experiment depicted in Fig. 5 clearly indicate that for the static networks — compared to the temporal one — for combinations of high seeding budget and low threshold, a static network is easily getting penetrated by the LTM diffusion process reaching almost 100% nodes existing in the network. This is not that surprising since in the static network that collapses all the links, there are more paths to reaching other nodes. Contrary to that, in the same

Table 3 Parameters used in the LTM in temporal networks experiment.

Symbol	Formula/Value	Description
θ	0.1	Edge removal threshold
λ	0.8	Forgetting function parameter
μ	$\mu \in \{0.05, 0.1, 0.15, 0.2, 0.25, 0.5\}$	Threshold of influence for LTM model
γ	$\gamma \in \{1, 5, 15, 25, 50\}$	Seeding budget (in %)

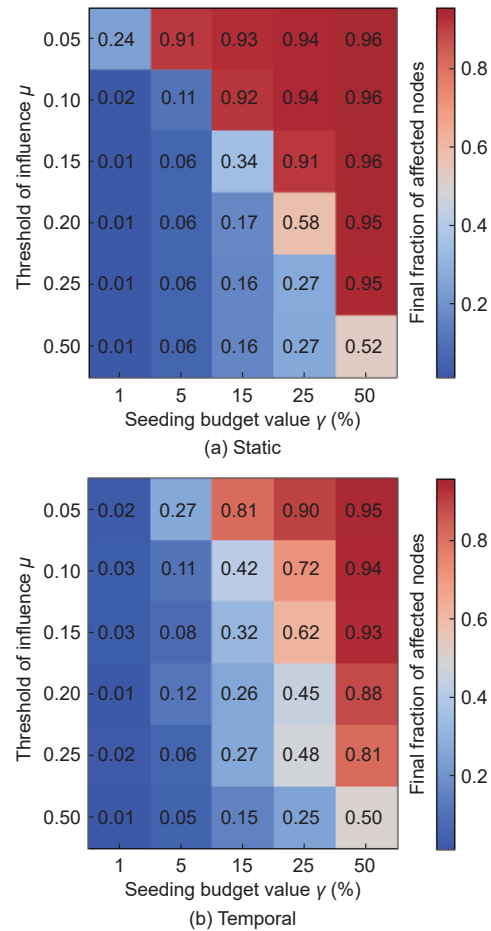


Fig. 5 Comparison of a final spread as a percentage of activated nodes for the LTM model of social influence for two settings: the temporal network based on the CogSNet model and the aggregated static network for the e-mail exchange data in a manufacturing company. The evaluated parameters are the seeding budget (γ) and the LTM threshold (μ).

regime of threshold values and percentage of initially activated nodes, the spread slows down in the temporal network built upon the CogSNet model. That can be caused by the absence of some potential links that do not appear in the later stage of the network evolution and that could not have been used for activation.

Yet, there is one interesting observation that one can make when looking at the combination of the LTM model parameters that are more challenging for the diffusion process (the central parts of Figs. 5a and 5b). Here, the CogSNet-modelled temporal network leads to a higher number of activations. It is due to the fact that

there are fewer links in the ego networks of activated nodes that had to be activated to change the state of the node. This means that the lower density of the network results in a higher number of activations in this regime. Finally, both approaches perform equally badly in the most difficult scenario (left sides of Figs. 5a and 5b).

4.3 Problem: Multilayer independent cascade model initialised with minimum dominating set

4.3.1 Problem formulation

The influence maximisation problem is well-known because of the work of Ref. [12] and adopts various forms^[7]. Studies have shown that the ultimate diffusion effectiveness depends not only on the spreading process itself and its parameters but also significantly on the network structure^[36–38]. In general, ranking methods for selecting seed nodes can be categorised into three groups: local, semi-local, and global^[39], differing in the scope of the network needed to analyze to determine the metric's value for a given node. Each of these classes varies in the trade-off between accuracy and computational complexity. Therefore, research teams often have to accept lower accuracy in identifying super-spreaders to obtain results within a reasonable time on large networks. Occasionally, one can leverage additional features to select the seed set, e.g., node labels, which are often available when analyzing data derived from social networks, as exemplified in Ref. [40]. However, some studies have shown that it is possible to determine (by analyzing the network's structure) whether computationally expensive methods can surpass basic algorithms, such as degree centrality^[41].

Maximising influence problem has long transcended the classical defined graphs as a set of nodes and edges. This experiment will focus on multilayer networks^[42]. By utilising the Network Diffusion framework, we will conduct a study to determine (in a minimal extent) whether network control methods are suitable for selecting initial nodes for propagation in graphs of this type. In order to build new seed selection strategies for the spreading processes in networks, we adopt the concept of driver nodes from the control theory. To control a system, one needs first to identify the minimum set of nodes (called driver nodes) that, when driven by distinct signals, can offer full control over the network. We argue that network spreading processes can be viewed as a soft version of control,

where control diffuses with a given probability of following certain rule(s). Thus, the intuition is that using driver nodes as seed nodes can result in a more efficient spread over the network, and indeed, it has been successfully employed for classical graphs^[17].

Networks with different characteristics feature different numbers of driver nodes. Once the driver nodes are identified, various centrality measures can be used to rank the driver nodes and use this final ranking list in the seed selection process. Experiments show that when driver nodes are used as the basis to build the ranking list, the spreading process is more effective and efficient when compared with the benchmarks. The first study that employs control-theoretic approaches for seed selection utilises the Minimum Dominating Set (MDS) to identify a candidate set of driver nodes^[17]. This method has been incorporated into the Network Diffusion framework and extended to work with multilayer networks.

MDS is an optimisation approach that determines the minimum dominating set of nodes in undirected networks^[16]. MDS is the smallest subset of nodes, such that every node of a network either belongs to this subset or is adjacent to at least one node in this set. Formally, a dominating set of a graph G is a subset D of the vertices of G , such that every vertex v of G is either in the set D or v has at least one neighbour that is in D .

Each driver node (or in the case of multilayer network: actor) can control its associated nodes (actors) independently, and each non-driver node (actor) is controllable if it is at least adjacent to one driver node (actor). In the MDS method, each node can exert control over its immediate neighbours at the same time, but the control does not propagate further. The driver nodes (actors) are chosen based on a minimal set of nodes (actors) that ensures each node (actor) in the network is either a driver node (actor) itself or directly connected to one. MDS has been applied in various contexts, such as characterising the perturbation of disease genes in the human regulatory network^[44] and identifying control variables in protein interaction networks^[45].

4.3.2 Experiment setup

In order to demonstrate the use of driver actors as the seeds in the spreading process, we run an experiment using AUCS^[46] and Lazega^[47] multilayer networks and apply an ICM^[12].

In the ICM we start by activating a set of nodes (seeds) which become active. Each active node (in the case of a multilayer network an actor) has one chance to activate each of its neighbours. It succeeds with some probability, known as activation (or propagation) probability. After one iteration, each active node becomes activated and cannot activate anyone anymore. At this point, it is good to note that in the case of multilayer networks actors, not nodes, are the subject of the diffusion (i.e., we care whether the actor is active or not, not nodes that just represent the actor on the particular layers). Therefore, one needs to define how to aggregate impulses from layers of the network influencing the actor. We adopt here the approach proposed originally for LTM extended for multiplex networks (described in Ref. [13]) called “protocol functions” (or strategies). In this study, we use two extreme ones: OR, i.e., an actor gets activated when it gets activated on at least one layer, and AND, i.e., an actor gets activated when it gets activated on all layers it is represented in. It is good to add that between them, there is an entire spectrum of all possible functions that one can invent.

For the first experiment, we use an activation probability of 10% and AND strategy. We test two different values of the seeding budget, i.e., 5% and 10%. To show how the MDS method performs in the context, we compare it with seed selection strategies based on Degree, Betweenness^[48], Closeness^[49], VoteRank^[50], and Berahmand’s centrality^[51]. They are adapted to work with multilayer networks as follows: Closeness, Betweenness, and Berahmand’s centrality, which are computed separately in each layer of the graph, and then final values for actors are obtained by

averaging scores from each layer. We modify VoteRank by replacing the Degree of an actor with neighbourhood size^[52]. Finally, MDS is obtained as a union of minimal dominating sets calculated on each layer of the network. We transform it into a ranking list by sorting according to the Degrees of actors. Averaged results depicting the efficiency of all evaluated methods can be found in Tables 4 and 5.

In order to show the dynamics of the spreading process triggered by actors belonging to MDS (sorted by degree centrality), we conduct a second simulation. For that, we utilize the AUCS network and multilayer ICM, with a 10% seeding budget, 50% activation probability, and OR strategy. The illustration of the obtained results is depicted in Fig. 6.

4.3.3 Results & discussion

The results for the AND strategy show that the MDS method performs at a similar level as other methods on the two selected datasets regardless of the seeding budget, and it takes a similar number of epochs to reach a stable state where no more new nodes are being activated (Tables 4 and 5). However, it should be noted that the parameters of the spreading model are used for illustration purposes only. The optimal ones are shown to depend on the network structural characteristics^[17] and can be optimised separately, but it is out of the scope of this study.

The results of spreading in the AUCS network using the OR strategy are shown in Fig. 6. In each of the layers, the number of activated nodes goes through a phase transition between epoch one and two resulting in a sudden increase in the number of activated nodes. This shows that MDS is a good strategy to activate many nodes early in the spread process. Again, used

Table 4 Average percentage of activated actors for multilayer ICM on AUCS network and average number of epochs needed for the model to reach a stable state when the simulation is stopped. The results for each seed selection method are averaged over 100 runs. The model is executed with 5% and 10% seeding budgets, 50% propagation probability, and AND strategy.

Seed selection method	Seeding budget 5%		Seeding budget 10%	
	Average activated actors $\pm$ std	Average number of epochs $\pm$ std	Average activated actors $\pm$ std	Average number of epochs $\pm$ std
Degree	(11.64 $\pm$ 3.17)%	2.82 $\pm$ 0.77	(20.90 $\pm$ 2.66)%	2.79 $\pm$ 0.50
Betweenness	(9.79 $\pm$ 2.51)%	2.54 $\pm$ 0.68	(18.11 $\pm$ 3.02)%	2.76 $\pm$ 0.55
Closeness	(8.79 $\pm$ 1.83)%	2.33 $\pm$ 0.60	(16.49 $\pm$ 2.43)%	2.46 $\pm$ 0.54
VoteRank	(10.98 $\pm$ 3.90)%	2.76 $\pm$ 0.91	(20.30 $\pm$ 3.81)%	2.88 $\pm$ 0.52
Berahmand’s centrality	(7.72 $\pm$ 2.55)%	2.10 $\pm$ 0.84	(13.56 $\pm$ 1.91)%	2.38 $\pm$ 0.58
MDS	(12.31 $\pm$ 3.15)%	2.84 $\pm$ 0.73	(19.46 $\pm$ 3.84)%	2.80 $\pm$ 0.51

Table 5 Average percentage of activated actors for multilayer ICM on Lazega network and average number of epochs needed for the model to reach a stable state when the simulation is stopped. The results for each seed selection method are averaged over 100 runs. The model was executed with 5% and 10% seeding budgets, 50% propagation probability, and AND strategy.

Seed selection method	Seeding budget 5%		Seeding budget 10%	
	Average activated actors $\pm$ std	Average number of epochs $\pm$ std	Average activated actors $\pm$ std	Average number of epochs $\pm$ std
Degree	$(79.17 \pm 17.74)\%$	6.01 ± 0.99	$(88.39 \pm 3.52)\%$	4.37 ± 0.52
Betweenness	$(78.03 \pm 15.53)\%$	6.02 ± 0.96	$(89.32 \pm 3.40)\%$	4.27 ± 0.44
Closeness	$(78.55 \pm 13.07)\%$	6.17 ± 0.90	$(89.10 \pm 3.35)\%$	4.43 ± 0.57
VoteRank	$(82.10 \pm 13.30)\%$	5.84 ± 0.90	$(89.03 \pm 3.40)\%$	4.44 ± 0.50
Berahmand's centrality	$(83.10 \pm 11.92)\%$	5.83 ± 1.02	$(89.13 \pm 3.19)\%$	4.31 ± 0.50
MDS	$(84.90 \pm 6.04)\%$	5.77 ± 0.85	$(89.28 \pm 3.28)\%$	4.28 ± 0.45

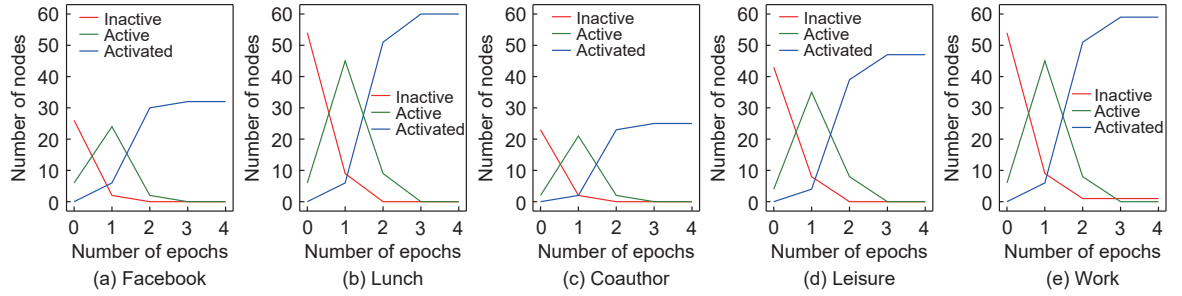


Fig. 6 Dynamics of ICM spreading in each layer of the AUCS network. The model is executed using MDS (degree centrality) as the seed selection method, with a 10% seeding budget, 50% propagation probability, and OR strategy. “Active” means that the node is active and has the potential to activate its neighbours; “Activated” denotes the node that is active but does not have the potential to activate its neighbours any more.

parameters are not optimised and are used for illustrative purposes.

4.4 Problem: Network epistemology model in temporal networks

4.4.1 Problem formulation

NEM is an approach to the formalisation of the social learning process, focusing on belief evolution and dissemination in the social network. Previous research on this model has explored various aspects, including the impact of community structure^[14], conformism, and polarisation^[53–56] on the effect of the learning process. By adapting the model to a temporal framework, it becomes better equipped to capture the ongoing social network dynamics, aligning it more closely with real-world scenarios^[15].

In its basic configuration, the model assumes the existence of a set of K agents organised within a temporal social network, each facing a bandit decision-making problem. At each iteration of the process, agents are presented with a choice between two available actions. The payoff associated with the first

action denoted as A , is common knowledge and remains constant at 0.5. In contrast, the payoff associated with the alternative action denoted as B , remains unknown to agents and deviates from the payoff of the action A by a specified value, denoted as ϵ .

Each agent’s decision relies on their individual belief level. The belief assigned to each agent is a value within the range of 0 to 1 interpreted as the probability that the given agent associates with the proposition that action B yields a superior payoff compared to action A .

Subsequent to their decision-making, agents who opt for action B engage in an experimentation phase, collecting evidence related to their actions. They execute the action B a defined N number of times, drawing from a Bernoulli distribution with a probability of success equal to $0.5 + \epsilon$.

Following this experimentation phase, agents update their belief levels using Bayes’ rule, calculating posterior probabilities based on their prior beliefs and the evidence they have accumulated. This updating process occurs in a synchronous manner across all

agents within the network. The simulation proceeds iteratively through each snapshot of the temporal network.

In the following experiment, we are going to compare the diffusion following NEM between the classical static approach to construct the underlying network topology and the temporal strategy with the CogSNet model^[18]. One can notice a similarity with the experiment described in Section 4.2. In fact, it exists, but the spreading model used here is much more complex than LTM. Therefore, the results between them are certainly not redundant.

4.4.2 Experiment setup

Implementation of the model with an extension allowing to run it in the temporal setting is located in `TemporalNetworkEpistemologyModel` class. Apart from previously discussed seeding properties, construction of the model instance requires two specific parameters described in the brief model definition, namely ϵ denoted as `epsilon`, and N denoted as `trials_nr`.

As stated above, in the experiment, we use two types of networks: static and dynamic. Both of them are constructed from a dataset on email exchange in manufacturing company^[35]. The number of experiments N that agents perform each iteration is equal to 1. We test different values of the seeding budget γ and values of ϵ , which stands for the difference between expected values of A and B actions. For both cases, we use the random seeding method. The value we compare between these configurations is the final number of actors voting for superior action B . The snapshots constituting the temporal network in this experiment are created with one day interval, resulting in 55 long sequences, with one snapshot for each following day of the underlying data. The parameters of NEM can be established based on the expert knowledge related to a specific sociological phenomenon or by a direct measurement of the underlying social learning process characteristic. In this example the fixed parameter $N = 1$ corresponds to one decision made daily by each agent during the simulation process. The configuration of CogSNet parameters is based on the work in Ref. [18], in which it is demonstrated to effectively match the ground truth data on subjects relations. The values of the experiments' outcomes are averaged across 100 iterations to reduce the random effects. Table 6 outlines

Table 6 Parameters used in the Network Epistemology Model in temporal networks experiment.

Symbol	Formula/Value	Description
θ	0.1	Edge removal threshold
λ	0.8	Forgetting function parameter
N	1	Number of experiments (trials) performed by an agent per iteration
ϵ	$\epsilon \in \{0.005, 0.01, 0.025, 0.05, 0.1\}$	Difference between expected values of A and B actions
γ	$\gamma \in \{1, 5, 15, 25, 50\}$	Seeding budget (in %)

the experiment parameters and their description.

4.4.3 Results & discussion

The outcomes of the experiment are shown in Fig. 7. The measure of efficiency of the explored configurations is expressed as the ratio of better, B

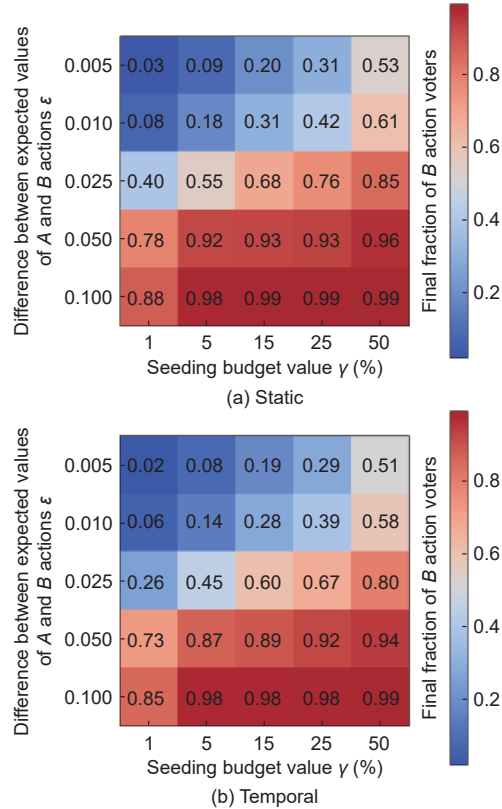


Fig. 7 Number of B action voters at the end of the spreading process compared for different levels of problem difficulty and various initial seeding budgets, that in this experiment corresponds to an initial number of B voters. The process outcomes are compared for two different approaches for network topology construction — a static network and a temporal network built using the CogSNet method.

action voters to all individuals. The observed results reveal a subtle difference between the two networks in the effectiveness of B action propagation. On average, in the temporal network, diffusion for each configuration reached a smaller or equal range compared to the static network. The most noticeable differences are in the test configurations, where the ϵ value was 0.025, where propagation in the static network covers a few to several percentage points more nodes.

4.5 Data used in experiments

As part of the experiments, we utilize several types of networks. Their basic parameters have been listed in Table 7. It is worth mentioning that the aucs-2 network is created with “lunch” and “facebook” layers to simulate the SIR and UA processes, respectively. The er-2 and sf-2 graphs are created using the NetworkX^[59] library, so that the layer responsible for disease transmission is sparser than the one responsible for spreading awareness about the epidemic. Implementation details can be found in the accompanying repository.

5 Limitations & Performance Study

In Section 4 of the paper, we briefly discuss limitations of the library and complement it with the performance analysis. Apart from the package functionalities described in Section 3, it is essential to address its constraints critically. They primarily stem from the adopted design assumptions and the relatively small size of the team involved in the implementation.

The most significant drawback of Network Diffusion is also its greatest advantage — implementation in Python. This results in significantly slower performance compared to if it had been coded in another compiled language. Although a part of the code has been provided in C language and is accessible as bindings to Python, we cannot adopt this approach in

the entire project due to the small team. Another package limitation is its support solely for discrete spreading processes (see Section 3). Though the framework is designed more like a set of interfaces to be used in the implementation of custom experiments, it does not include many concrete spreading models that work “out of the box”. Comparing that to NDlib, which consists of dozens of pre-defined models, one can find that state to be a limitation. Finally, it is worth noting the absence of a user interface, which may prove impossible for individuals lacking programming skills, leaving them compelled to resort to tools like NetLogo^[60].

In order to assess the performance of the Network Diffusion, we utilize all four models described in the article. The domain chosen to express complexity is the experiment duration as a function of the network size, expressed in the number of actors. We evaluate the following set of parameters:

- Number of simulation steps — 200 (regardless of whether a stable state is achieved earlier or not);
- Evaluation is performed on two-layer multiplex Erdős-Rényi networks with an edge creation probability of 0.1 and the number of actors from range 10 to 1000; and
- Each experiment is repeated 10 times.

The results are presented in Fig. 8. As can be seen, the most computationally expensive models are NEM and LTM, which significantly differ from ICM and SIR-UA. It is also worth noting that the computation time is stable — a standard deviation (a blurry region surrounding curves) is insignificant.

Considering the future of the package, we undoubtedly are going to develop and maintain it. Having defined the framework, it is natural to enhance its contents by new spreading models and functions regarding both temporal and multilayer networks. We are also going to implement machine learning based methods for the identification of the most influencing

Table 7 Networks used in experiments with their basic parameters shortlisted.

Name	Number of layers	Number of actors	Number of nodes	Number of edges	Note
aucs	5	61	224	620	AUCS network ^[46]
aucs-2	2	60	120	317	AUCS network ^[46]
lazega	3	71	212	1659	Lazega Law Firm network ^[47]
er-2	2	1000	2000	27 451	Erdős-Rényi network ^[57]
sf-2	2	1000	2000	3357	Scale-free network ^[58]
manuf	1	151	151	20 000	E-mail exchange ^[35]

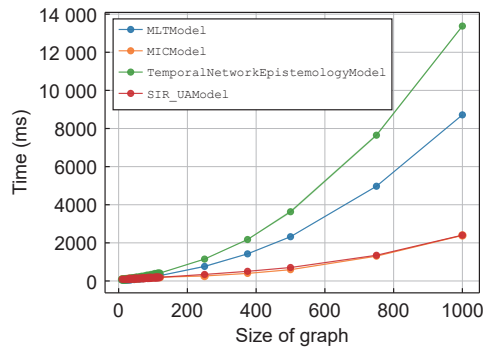


Fig. 8 Complexity of spreading models used in experiments; evaluation is performed on two-layer multiplex Erdős-Rényi networks.

actors. Nonetheless, there is another path that is especially interesting to us — matching the theoretical spreading model to the observed phenomena “in the wild”. That will be addressed within the Network Diffusion or at least with its support.

6 Conclusion

Developing and making publicly available comprehensive software frameworks that support replicability and dissemination of research effort is a key and very much required component of scientific enquiry. In order to support the research community, we show in this paper an extended version of the Network Diffusion framework. Its initial concept is presented in Ref. [9] and has now been significantly enriched to encompass both temporal and multilayer networks, as well as various methods and spreading processes that can propagate over different types of graphs.

The models covered in this study include multilayer networks and temporal networks. Additionally, we adopt the node centrality measures for multilayer networks. Spreading models covered in this paper include SIR-UA, temporal LTM, temporal NEM, and multilayer ICM. Presented spreading models, which have been included in the framework’s extension, are accompanied by experiments to show how the newly developed parts of the library can be utilized. They also show the interesting future direction of our research that we would like to investigate in the future fully.

As discussed above, the Network Diffusion framework is complementary to other software solutions available in the field of data science. We

believe that its comprehensiveness and ease of use make it accessible not only for computer scientists, but also for researchers from other domains who are interested in understanding spread in the systems they work on.

Authors Contribution

Conceptualization (Michał Czuba and Piotr Bródka), data curation (Michał Czuba, Yu-Xuan Qiu, Mateusz Nurek, and Radosław Michalski), formal analysis (All), funding acquisition (Piotr Bródka and Radosław Michalski), investigation (All), methodology (Michał Czuba, Piotr Bródka, Radosław Michalski, and Katarzyna Musiał), project administration (Michał Czuba), resources (All), software (Michał Czuba, Damian Serwata, Mateusz Nurek, Mingshan Jia, and Yu-Xuan Qiu), supervision (Piotr Bródka), validation (All), visualisation (Michał Czuba, Damian Serwata, Mateusz Nurek, Mingshan Jia, and Yu-Xuan Qiu), writing – original draft (All), writing – review & editing (Michał Czuba and Piotr Bródka).

Acknowledgment

This research was partially supported by the National Science Centre, Poland (Nos. 2022/45/B/ST6/04145 and 2021/41/B/HS6/02798), and the EU under the Horizon Europe (No.101086321 OMINO). Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the National Science Centre, EU, or the European Research Executive Agency.

References

- [1] F. Menczer, S. Fortunato, and C. A. Davis, *A First Course in Network Science*. Cambridge, UK: Cambridge University Press, 2020.
- [2] R. Pastor-Satorras and A. Vespignani, Epidemic spreading in scale-free networks, *Phys. Rev. Lett.*, vol. 86, no. 14, pp. 3200–3203, 2001.
- [3] M. Nurek, R. Michalski, O. Lizardo, and M. A. Rizoiu, Predicting relationship labels and individual personality traits from telecommunication history in social networks using hawkes processes, *IEEE Access*, vol. 11, pp. 8492–8503, 2023.
- [4] S. Forouzandeh, K. Berahmand, R. Sheikhpour, and Y. Li, A new method for recommendation based on embedding spectral clustering in heterogeneous networks (RESCHet), *Expert Syst. Appl.*, vol. 231, p. 120699, 2023.
- [5] M. Rostami, U. Muhammad, S. Forouzandeh, K. Berahmand, V. Farrahi, and M. Oussalah, An effective explainable food recommendation using deep image clustering and community detection, *Intell. Syst. Appl.*, vol. 16, p. 200157, 2022.

- [6] C. Zhong, S. M. Arisona, X. Huang, M. Batty, and G. Schmitt, Detecting the dynamics of urban structure through spatial network analysis, *Int. J. Geogr. Inf. Sci.*, vol. 28, no. 11, pp. 2178–2199, 2014.
- [7] S. S. Singh, D. Srivastva, M. Verma, and J. Singh, Influence maximization frameworks, performance, challenges and directions on social network: A theoretical study, *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 9, pp. 7570–7603, 2022.
- [8] P. Brodka, K. Musial, and J. Jankowski, Interacting spreading processes in multilayer networks: A systematic review, *IEEE Access*, vol. 8, pp. 10316–10341, 2020.
- [9] M. Czuba and P. Brodka, Simulating spreading of multiple interacting processes in complex networks, in *Proc. 2022 IEEE 9th Int. Conf. Data Science and Advanced Analytics (DSAA)*, Shenzhen, China, 2022, pp. 1–10.
- [10] A. L. Barabási, *Network Science*. Cambridge, UK: Cambridge University Press, 2016.
- [11] M. Granovetter, Threshold models of collective behavior, *Am J Sociol*, vol. 83, no. 6, pp. 1420–1443, 1978.
- [12] D. Kempe, J. Kleinberg, and É. Tardos, Maximizing the spread of influence through a social network, in *Proc. 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Washington, DC, USA, 2003, pp. 137–146.
- [13] Y. D. Zhong, V. Srivastava, and N. E. Leonard, Influence spread in the heterogeneous multiplex linear threshold model, *IEEE Trans. Control Netw. Syst.*, vol. 9, no. 3, pp. 1080–1091, 2022.
- [14] K. J. S. Zollman, The communication structure of epistemic communities, *Philos. Sci.*, vol. 74, no. 5, pp. 574–587, 2007.
- [15] R. Michalski, D. Serwata, M. Nurek, B. K. Szymanski, P. Kazienko, and T. Jia, Temporal network epistemology: On reaching consensus in a real-world setting, *Chaos*, vol. 32, no. 6, p. 063135, 2022.
- [16] J. C. Nacher and T. Akutsu, Dominating scale-free networks with variable scaling exponent: Heterogeneous networks are not difficult to control, *New J. Phys.*, vol. 14, no. 7, p. 073005, 2012.
- [17] A. Sadaf, L. Mathieson, P. Bródka, and K. Musial, Maximising influence spread in complex networks by utilising community-based driver nodes as seeds, in *Information Management and Big Data*, J. A. Lossio-Ventura, J. Valverde-Rebaza, E. Díaz, and H. Alatrasta-Salas, eds. Lima, Peru: Springer, 2023, pp. 126–141.
- [18] R. Michalski, B. K. Szymanski, P. Kazienko, C. Lebiere, O. Lizardo, and M. Kulisiewicz, Social networks through the prism of cognition, *Complexity*, vol. 2021, p. 4963903, 2021.
- [19] W. Van den Broeck, C. Gioannini, B. Goncalves, M. Quagiotto, V. Colizza, and A. Vespignani, The GLEaMviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale, *BMC Infect Dis*, vol. 11, no. 1, p. 37, 2011.
- [20] G. Rossetti, L. Milli, S. Rinzivillo, A. Sîrbu, D. Pedreschi, and F. Giannotti, NDlib: A python library to model and analyze diffusion processes over complex networks, *Int. J. Data Sci. Anal.*, vol. 5, no. 1, pp. 61–79, 2018.
- [21] S. Widgren, P. Bauer, R. Eriksson, and S. Engblom, SimInf: An R package for data-driven stochastic disease spread simulations, *J. Stat. Softw.*, vol. 91, no. 12, pp. 1–42, 2019.
- [22] F. P. Alvarez, P. Crépey, M. Barthélemy, and A. J. Valleron, Sispread: A software to simulate infectious diseases spreading on contact networks, *Methods Inf. Med.*, vol. 46, no. 1, pp. 19–26, 2007.
- [23] J. V. Douglas, S. Bianco, S. Edlund, T. Engelhardt, M. Filter, T. Günther, K. Hu, E. J. Nixon, N. L. Sevilla, A. Swaid, et al., STEM: An open source tool for disease modeling, *Health Secur.*, vol. 17, no. 4, pp. 291–306, 2019.
- [24] S. M. Jenness, S. M. Goodreau, and M. Morris, EpiModel: An R package for mathematical modeling of infectious disease over networks, *J. Stat. Softw.*, vol. 84, p. 8, 2018.
- [25] A. A. Toda, Susceptible-Infected-Recovered (SIR) dynamics of COVID-19 and economic impact, arXiv preprint arXiv: 2003.11221, 2020.
- [26] P. Wątroba and P. Bródka, Influence of information blocking on the spread of virus in multilayer networks, *Entropy*, vol. 25, no. 2, p. 231, 2023.
- [27] D. K. Chu, E. A. Akl, S. Duda, K. Solo, S. Yaacoub, H. J. Schunemann, Schunemann, and COVID-19 Systematic Urgent Review Group Effort (SURGE) study authors, physical distancing, face masks, and eye protection to prevent person-to-person transmission of SARS-CoV-2 and COVID-19: A systematic review and meta-analysis, *Lancet*, vol. 395, no. 10242, pp. 1973–1987, 2020.
- [28] D. J. McGrail, J. Dai, K. M. McAndrews, and R. Kalluri, Enacting national social distancing policies corresponds with dramatic reduction in COVID19 infection rates, *PLoS One*, vol. 15, no. 7, p. e0236619, 2020.
- [29] M. Alene, L. Yismaw, M. A. Assemie, D. B. Ketema, B. Mengist, B. Kassie, and T. Y. Birhan, Magnitude of asymptomatic COVID-19 cases throughout the course of infection: A systematic review and meta-analysis, *PLoS One*, vol. 16, no. 3, p. e0249090, 2021.
- [30] P. Holme and J. Saramaki, Temporal networks, *Phys. Rep.*, vol. 519, no. 3, pp. 97–125, 2012.
- [31] R. Michalski, T. Kajdanowicz, P. Bródka, and P. Kazienko, Seed selection for spread of influence in social networks: Temporal vs. static approach, *New Gener. Comput.*, vol. 32, nos. 3&4, pp. 213–235, 2014.
- [32] R. Michalski and P. Kazienko, Maximizing social influence in real-world networks — the state of the art and current challenges, in *Propagation Phenomena in Real World Networks*, D. Król, D. Fay, and B. Gabrys, eds. Cham, Switzerland: Springer, 2015, pp. 329–359.
- [33] I. Scholtes, N. Wider, R. Pfitzner, A. Garas, C. J. Tessone, and F. Schweitzer, Causality-driven slow-down and speed-up of diffusion in non-Markovian temporal networks, *Nat. Commun.*, vol. 5, no. 1, p. 5024, 2014.

- [34] F. Karimi and P. Holme, Threshold model of cascades in empirical temporal networks, *Physica A*, vol. 392, no. 16, p. 3476–3483, 2013.
- [35] M. Nurek and R. Michalski, Combining machine learning and social network analysis to reveal the organizational structures, *Appl. Sci.*, vol. 10, no. 5, p. 1699, 2020.
- [36] R. Albert and A. L. Barabasi, Statistical mechanics of complex networks, *Rev. Mod. Phys.*, vol. 74, no. 1, pp. 47–97, 2002.
- [37] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. U. Hwang, Complex networks: Structure and dynamics, *Phys. Rep.*, vol. 424, nos. 4&5, pp. 175–308, 2006.
- [38] Z. Liu, X. Wu, and P. M. Hui, An alternative approach to characterize the topology of complex networks and its application in epidemic spreading, *Front. Comput. Sci. China*, vol. 3, no. 3, pp. 324–334, 2009.
- [39] L. Lü, D. Chen, X. L. Ren, Q. M. Zhang, Y. C. Zhang, and T. Zhou, Vital nodes identification in complex networks, *Phys. Rep.*, vol. 650, pp. 1–63, 2016.
- [40] S. Forouzandeh, A. Sheikhhahmadi, A. R. Aghdam, and S. Xu, New centrality measure for nodes based on user social status and behavior on Facebook, *Int. J. Web Inf. Syst.*, vol. 14, no. 2, pp. 158–176, 2018.
- [41] K. Berahmand, N. Samadi, and S. M. Sheikholeslami, Effect of rich-club on diffusion in complex networks, *Int. J. Mod. Phys. B*, vol. 32, no. 12, p. 1850142, 2018.
- [42] M. E. Dickison, M. Magnani, and L. Rossi, *Multilayer Social Networks*. Cambridge, UK: Cambridge University Press, 2016.
- [43] A. Sadaf, L. Mathieson, and K. Musial, Effects of global and local network structure on number of driver nodes in complex networks, in *Cyber Security and Social Media Applications*, S. T. Özyer and B. Kaya, eds. Cham, Switzerland: Springer, 2023, pp. 81–98.
- [44] B. Wang, L. Gao, Q. Zhang, A. Li, Y. Deng, and X. Guo, Diversified control paths: A significant way disease genes perturb the human regulatory network, *PLoS One*, vol. 10, no. 8, p. e0135491, 2015.
- [45] S. Wuchty, Controllability in protein interaction networks, *Proc. Natl. Acad. Sci. USA*, vol. 111, no. 19, pp. 7156–7160, 2014.
- [46] L. Rossi and M. Magnani, Towards effective visual analytics on multiplex and multilayer networks, *Chaos Solitons Fractals*, vol. 72, pp. 68–76, 2015.
- [47] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, New specifications for exponential random graph models, *Sociological Methodology*, vol. 36, no. 1, pp. 99–153, 2006.
- [48] U. Brandes, A faster algorithm for betweenness centrality, *J. Math. Sociol.*, vol. 25, no. 2, pp. 163–177, 2001.
- [49] L. C. Freeman, Centrality in social networks conceptual clarification, *Soc. Netw.*, vol. 1, no. 3, pp. 215–239, 1978.
- [50] J. X. Zhang, D. B. Chen, Q. Dong, and Z. D. Zhao, Identifying a set of influential spreaders in complex networks, *Sci. Rep.*, vol. 6, p. 27823, 2016.
- [51] K. Berahmand, A. Bouyer, and N. Samadi, A new centrality measure based on the negative and positive effects of clustering coefficient for identifying influential spreaders in complex networks, *Chaos Solitons Fractals*, vol. 110, p. 41–54, 2018.
- [52] P. Bródka, J. Jankowski, and R. Michalski, Sequential seeding in multilayer networks, *Chaos*, vol. 31, no. 3, p. 033130, 2021.
- [53] C. O'Connor and J. Weatherall, Scientific polarization, *Eur. J. Philos. Sci.*, vol. 8, no. 3, p. 855–875, 2018.
- [54] J. Weatherall and C. O'Connor, Do as I say, not as I do, or, conformity in scientific networks, arXiv preprint arXiv: 1803.09905, 2019.
- [55] J. Weatherall and C. O'Connor, Endogenous epistemic factionalization: A network epistemology approach, arXiv preprint arXiv: 1812.08131.
- [56] J. O. Weatherall, C. O'Connor, and J. Bruner, How to beat science and influence people: Policymakers and propaganda in epistemic networks, *Br. J. Philos. Sci.*, vol. 71, no. 4, pp. 1157–1186, 2020.
- [57] P. Erdős and A. Rényi, On the evolution of random graphs, *Publ. Math. Inst. Hung. Acad. Sci.*, vol. 5, no. 1, pp. 17–60, 1960.
- [58] B. Bollobás, C. Borgs, J. Chayes, and O. Riordan, Directed scale-free graphs, in *Proc. 14th Annu. ACM-SIAM Symp. Discrete Algorithms*, Baltimore, MD, USA, 2003, pp. 132–139.
- [59] A. A. Hagberg, D. A. Schult, and P. J. Swart, Exploring network structure, dynamics, and function using networkx, in *Proc. 7th Python in Science Conf.*, Pasadena, CA, USA, 2008, pp. 11–15.
- [60] U. Wilensky, NetLogo: Center for connected learning and computer-based modeling, <http://ccl.northwestern.edu/netlogo>, 2023.



Michał Czuba is a PhD candidate at Wrocław University of Science and Technology, Poland, in the discipline of information and communication technology with a focus on computational network science, namely problems of influence maximisation and phenomena spreading in multilayer networks. He also

has industrial expertise in machine learning, computer vision systems, and MLOps engineering in projects concerning the commercialisation of research outcomes.



Mateusz Nurek is a PhD candidate at Wrocław University of Science and Technology, Poland. His research area includes network science and machine learning. He is primarily interested in using computational intelligence to study social aspects; therefore, his current research focuses on problems, such as the

classification of human relationships or predicting personality traits based on communication patterns.



Damian Serwata is a PhD candidate at Wrocław University of Science and Technology, Poland. His research focus is concentrated on leveraging network science models and tools to study complex social systems and their adaptive abilities. Currently, he is devoted to research on the social learning process, with regards to the investigation of different approaches to represent social structures, bringing the models closer to reality.



Yu-Xuan Qiu presently serves as a postdoctoral research assistant at University of Technology Sydney, Australia. He received the BEng and Meng degrees in computer science from Shenzhen University, China in 2015 and 2018, respectively, and the PhD degree from University of Technology Sydney, Australia in 2023. His primary research interest is graph data mining and management.



Mingshan Jia is a lecturer at University of Technology Sydney, Australia. He received the BEng degree in information engineering from Xi'an Jiaotong University, China in 2008, the MEng degree in information and telecommunication systems from University of Technology of Troyes, France in 2011, and the PhD degree from University of Technology Sydney, Australia in 2022. Controllability of networks, data mining, and machine learning applications in social networks belong to his main research interests.



Radosław Michalski is an associate professor at Department of Artificial Intelligence, Wrocław University of Science and Technology, Poland, where he co-leads the Network Science Lab and leads the Blockchain Exploration Research Group (BERG). His research interests include social influence, diffusion processes in complex networks, and machine learning. He has co-authored more than 50 publications in these areas.



Katarzyna Musiał received the MEng degree in computer science from Wrocław University of Science and Technology, Poland in 2006, the another MEng degree in software engineering from the Blekinge Institute of Technology, Sweden in 2006, and the PhD degree from Wrocław University of Science and Technology, Poland in 2009. In the same year, she was appointed as a senior visiting research fellow at Bournemouth University, UK, where she has been a lecturer in informatics since 2010. She joined King's College London as a lecturer in computer science in 2011. In 2015, she returned to Bournemouth University, UK, where she was an associate professor of computing, as well as the head of the SMART Technology Research Group and a member of the Data Science Initiative. In 2017, she moved to Australia and started working as a professor of network science at Data Science Institute, University of Technology Sydney, Australia, where she co-leads the Complex Adaptive Systems Lab. She is particularly interested in social networks, graph controllability, and complex systems.



Piotr Bródka is an associate professor at Department of Artificial Intelligence, Wrocław University of Science and Technology, Poland. He received the MEng degree in computer science from Wrocław University of Technology, Poland in 2008, and the PhD degree in late 2012. In 2012, he also received another MEng degree in computer science from Blekinge Institute of Technology, Sweden. In 2020, he received a Habilitation (DSc) in information and communication technology. He was a visiting scholar at Stanford University, Australia in 2013, and a visiting professor at University of Technology Sydney, Australia in 2018 and 2019. He has authored over 100 scholarly and research articles on a variety of areas related to complex networks and computational network science, focusing on the extraction and dynamics of communities within social networks, spreading processes in complex networks, and the analysis of multilayer networks.

4.3 Peer Interaction Dynamics and Second Language Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social Network Analysis

Authors: Michał Paradowski, Nicole Whitby, Michał Czuba, Piotr Bródka

Published at: Language Learning; December 2024

MNiSW points: 200 (IF 4.2)

Citations: Web of Science: 9

(as of 6th March 2026) Scopus: 12

Google Scholar: 24

CRedit:

Methodology

Software

Formal analysis

Data curation

Writing – original draft

Visualisation



EMPIRICAL STUDY

Peer Interaction Dynamics and Second Language Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social Network Analysis

Michał B. Paradowski ^a, Nicole Whitby ^b,
Michał Czuba ^c and Piotr Bródka ^c

^aUniversity of Warsaw ^bIndependent Researcher ^cWrocław University of Science and Technology

CRediT author statement – **Michał B. Paradowski**: conceptualization (lead), funding acquisition, methodology (equal), formal analysis (equal), supervision, validation, writing – original draft (lead), review and editing (lead). **Nicole Whitby**: investigation, data curation (equal), writing – original draft (supporting), review and editing (supporting). **Michał Czuba**: data curation (equal), formal analysis (equal), methodology (equal), software (equal), visualization (equal), writing – original draft (supporting). **Piotr Bródka**: data curation (supporting), formal analysis (equal), methodology (equal), software (equal), visualization (equal), writing – original draft (supporting), review and editing (supporting).

A one-page Accessible Summary of this article in nontechnical language is freely available in the Supporting Information online and at <https://oasis-database.org>

We thank the student participants for taking the time to fill out the questionnaires, as well as the course coordinators for distributing these and facilitating access to coded pseudonymized data. The authors are grateful to the audiences of the conferences “Second Language Learning and Teaching: Taking Stock and Looking Ahead,” the 34th International Conference on Second/Foreign Language Acquisition, the 32nd Conference of the European Second Language Association (EuroSLA), “Residence Abroad and Language Learning: Where Are We Now?,” the 75th Georgetown University Round Table (GURT) “Education Abroad: Language, Learners, and Communities,” and AAAL 2024 for their insightful feedback; to the four anonymous reviewers of this manuscript for their helpful commentary and bibliographical suggestions; and to Abdullah Alamer, Ali H. Al-Hoorie, Breno Silva, Bronson Hui, Chris Nicklin, Hassan Khajavy, Joe Vitta, Luke Plonsky, and Stefan Th. Gries for discussion. Last but not least, we thank Jill Bowie, Mary Akbary, and Sarah Grey for the meticulous copyediting. All the usual disclaimers apply.

Correspondence concerning this article should be addressed to Michał B. Paradowski, Institute of Applied Linguistics, University of Warsaw, ul. Dobra 55, PL-00-312 Warsaw, Poland. Email: m.b.paradowski@uw.edu.pl

The handling editor for this article was Annarita Magliacane.

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

Abstract: Using computational Social Network Analysis (SNA), this longitudinal study investigates the development of the interaction network and its influence on the second language (L2) gains of a complete cohort of 41 U.S. sojourners enrolled in a 3-month intensive study-abroad Arabic program in Jordan. Unlike extant research, our study focuses on students' interactions with alma mater classmates, reconstructing their complete network, tracing the impact of individual students' positions in the social graph using centrality metrics, and incorporating a developmental perspective with three measurement points. Objective proficiency gains were influenced by predeparture proficiency (negatively), multilingualism, perceived integration of the peer learner group (negatively), and the number of fellow learners speaking to the student. Analyses reveal relatively stable same-gender cliques, but with changes in the patterns and strength of interaction. We also discuss interesting divergent trajectories of centrality metrics, L2 use, and progress; predictors of self-perceived progress across skills; and the interplay of context and gender.

Keywords computational social network analysis (SNA); study abroad; L2 Arabic; peer interaction; complete/whole (sociocentric) network approach

Introduction

By 2025, 8 million students globally are estimated to take part in study abroad (SA) programs (Gürüz, 2011). One of the most commonly listed goals of participating in these is improving target-language (TL, or L2)¹ competence. The experience of potential immersion in an L2-speaking community coupled with the opportunity of engaging in meaningful communication is commonly believed to be conducive to accelerated progress. However, such sojourns do not always lead to improved proficiency. To a certain extent this is because second language acquisition (SLA) is significantly affected by learners' social network structures and the dynamics of their communication. Yet, there is a dearth of studies aiming to properly explain these mechanisms, or to track sojourners' language use and development over several points in time.

This work was carried out as part of National Science Centre of Poland projects SONATA-BIS No. 2016/22/E/HS2/00034 (<https://peerlang.ils.uw.edu.pl>) and Opus No. 2022/45/B/ST6/04145 (<https://multispread.pwr.edu.pl>), and was partially supported by the Polish Ministry of Education and Science within the program "International Projects Co-Funded" and by the EU under Horizon Europe OMINO grant No. 101086321 (<https://ominoproject.eu>). The first author also acknowledges support from the University of Warsaw's "Excellence Initiative – Research University" (IDUB) program Priority Research Area IV "The Humanities: Crossing Borders, Extending Capabilities" as well as from COST Actions 15109 (COSTNET), 15130 (SAREP), 18232 (MAT-DYN-NET), 19102 (LITHME), 20115 (ENIS), and 21114 (CLILNetLE).

One line of inquiry that has attempted to explain the varied picture has investigated students' social networks as a factor conditioning their interactions and, subsequently, progress. However, most of the extant studies have i) focused on sojourners' interactions with native speakers of the TL, passing over their communication with other interlocutors (despite the fact that these conversations oftentimes constitute the majority of their linguistic experience), ii) only investigated participants' egocentric networks (asking individual learners to provide information about their social networks, without collecting corresponding information from their peers), iii) if at all, used solely global metrics of the networks extracted, and iv) only measured presojourn proficiency and one timepoint of post-stay gains.

This study analyzes the longitudinal development of the SA social interaction network and its influence on L2 gains of 41 U.S. sojourners enrolled in a 3-month intensive study-abroad Arabic program in Jordan. Unlike most previous study abroad social network research, we i) focus on students' interactions with classmates who share their home institution as well as with other agents, ii) obtain information from all of the students, thus permitting a reconstruction of a (nearly) full social graph, iii) trace the impact of each individual student's position in the network using centrality metrics, and iv) include a dynamic developmental perspective with three measurement points at 4-week intervals each, gauging the extent to which changes in the interaction networks translate to changes in progress along a range of dimensions. The approach contributes novel methodology and rigorous insights into the dynamics of study abroad SLA, and offers tangible recommendations for SA program stakeholders.

Background Literature

Role of Study Abroad Sojourns in L2 Development

Second language (L2) acquisition during study abroad has been a burgeoning field of inquiry over the last three decades. Since the seminal 1995 publication of *Second Language Acquisition in a Study Abroad Context* by Barbara Freed, SA has been recognized as a setting that has the potential to expose students to cultural and language learning experiences that lead to learning outcomes substantially different from the traditional at-home classroom settings (R. Mitchell et al., 2020). The experience of potential immersion in a community speaking the target language, coupled with the opportunity to engage in meaningful L2 communication, is commonly believed to be conducive to accelerated progress. The benefits have been argued to come from socialization in L2 communities of practice and the nature of the social relationships developed in these contexts (Dewey, 2017; Diao & Maa, 2019; Duff, 2019).

However, despite “folk beliefs” (Howard, 2011, p. 72) in the “immersion myth” (Jackson, 2017, p. 3) that learners’ mere presence in a TL-speaking country will automatically facilitate language acquisition virtually by “osmosis” (Davidson, 2010, p. 10), not all benefit equally from SA sojourns, and considerable variation has been evinced in students’ linguistic attainment (Atsuzawa-Windley & Noguchi, 1995; Barron, 2002; Chamot et al., 2021; Collentine & Freed, 2004; DeKeyser, 1991, 2014; Dewey, 2004; Dewey et al., 2014; Díaz-Campos, 2004; Hasegawa, 2019; Hendrickson, 2018; Isabelli-García et al., 2018; Jackson, 2017; Kinginger, 2009, 2011; Kinginger & Belz, 2005; Kinginger & Blattner, 2008; Kubota, 2016; Lafford & Uscinski, 2014; Marriot, 1995; McManus, 2019; Pérez-Vidal, 2014; Regan, 1995; Sanz & Morales-Front, 2018; Segalowitz et al., 2004; C. Wang, 2010). The results of SA studies have proven disparate enough for some scholars to conclude that “if there is any consistent finding in this research, it is of inconsistency” (Kinging, 2013, p. 7; see also Tullock & Ortega, 2017, for a scoping review).

While some authors have reported benefits of SA for gains in oral proficiency (e.g., Wright, 2018, for L2 Chinese; Juan-Garau, 2018, for L2 English; Freed, 1995, Lapkin et al., 1995, and Magnan & Back, 2007, for L2 French; R. Mitchell et al., 2017, for L2 French and Spanish; DeKeyser, 2010, Hernández, 2010, Huensch & Tracy-Ventura, 2017, and Isabelli-García, 2010, for L2 Spanish; Brecht et al., 1995, for L2 Russian), others have shown that even homestay placements may lead to facility only in predictable formulaic “chitchat” (e.g., Segalowitz & Freed, 2004; Shiri, 2015, p. 17), and yet other findings have downright refuted a positive relationship between linguistic gains and the amount of L2 use during SA (Baker-Smemoe et al., 2014; Miller & Ginsberg, 1995). SA is often credited with affordances to develop interlanguage pragmatics (e.g., Bataller, 2010; Iwasaki, 2010), though most learners have been found to demonstrate only incipient or partial acquisition of pragmatic constraints (Pérez Vidal & Shively, 2019; Sánchez Hernández, 2022); and it has demonstrated benefits for other variable and regional features of the L2 (Geeslin et al., 2013; Kennedy Terry, 2017; Knouse, 2012; Pozzi & Bayley, 2021; Regan et al., 2009). In turn, results from studies investigating the acquisition of specific grammatical, lexical, or phonological features of the L2 by SA as compared with at-home learners have presented a much more opaque picture, with relative gains insignificant or not across the board (Arnett, 2013; Collentine, 2004; Freed et al., 2003; Grey et al., 2015; Isabelli-García, 2010; McManus & Mitchell, 2015; R. Mitchell et al., 2017; Mora, 2007; Serrano et al., 2012; Steinwiddler, 2016).

This complex picture necessitates a shift from a more traditional “taken-for-granted” conceptualization of the SA learning context—one in which the setting is perceived “as a monolithic physical location where all students have equal access to all aspects of the language at all times” (Devlin, 2019, pp. 123–124; see also Coleman, 2015)—towards a more nuanced approach accounting for the dynamic nature of learner interactions. Hence numerous authors have been reiterating calls for additional research in this area (Borràs & Llanes, 2019; Isabelli-García et al., 2018; Issa et al., 2020; Marijuan & Sanz, 2018).

Social Networks and (Study-Abroad) L2 Development

The interaction-based research of the 1970s to the mid-1990s focused almost exclusively on negotiation for meaning in specially arranged “laboratory settings” in (English as a) foreign language contexts (Spada & Lightbown, 2009, pp. 157–163), although there was also “interaction analysis” research aimed at observing the patterns of teacher and student behaviors in the classroom (Chaudron, 1988). Both of these strands of research were primarily descriptive in nature (Spada & Lightbown, 2009, p. 162). However, the last two decades of research on L2 learning have seen a progression towards numerous diverse models and theories making explicit allowance for the indissoluble link between the process of language acquisition and social interactions (Atkinson, 2002; Duff, 2007, 2019; Lee et al., 2009; R. Mitchell et al., 2019; Tarone, 2007; Zuengler & Miller, 2006). This has taken place in line with the broader “growing recognition of the need to expand the theoretical scope and the database of SLA research ... to move beyond a narrowly cognitive orientation to include the impact of social factors” (Tarone, 2009, p. 41) as well as situational and contextual aspects (see, e.g., Atkinson, 2002; Geeslin & Long, 2014; Ortega, 2011; Regan, 2013). Such a research direction clearly treats language acquisition as a process (a) grounded in the activities (social interactions and relationships) in which language users engage as social beings, and (b) shaped by those activities (to the extent of affecting the sequence of interlanguage development; Tarone & Liu, 1995).

Particularly pertinent here is the interactionist approach, one of whose basic tenets is a focus on how learners use their linguistic environment to build their knowledge of the L2, taking into account the following:

- *input* (exposure to primary linguistic data that feeds into the language acquisition device; see, e.g., Lightbown & Spada, 2021),

- *output* (production, argued to develop fluency, de Bot, 1996, and to raise metalinguistic reflection and awareness, Lapkin et al., 1995; Swain, 1993), and
- direct and indirect feedback on output (Swain, 1993) via *negotiation for meaning* (Long's 1996 interaction hypothesis) and the "communicative pressure" of *interaction* (Gass, 2003, p. 224; Gass, 2010, p. 2; Mackey, 1999), which may facilitate the noticing of crucial features in the input.

Accordingly, an increasingly fruitful recent line of SA research has foregrounded the importance of exploring learner interactions by harnessing social network theory, with the hope of better understanding how students communicate and learn in both formal and informal social contexts (Carhill-Poza & Kurata, 2020; Dewey, 2017; Dewey, Belnap, & Hillstrom, 2013; Dewey, Ring, et al., 2013; Freed et al., 2004; Hasegawa, 2019; Kennedy Terry, 2017, 2022a, 2022b; Kurata, 2007, 2012; Li & Gong, 2022; R. Mitchell, 2023; R. Mitchell et al., 2015, 2017; Paradowski et al., 2021a, 2021b, 2022; Pérez-Vidal & Sanz, 2023; Pozzi, 2021; Pozzi & Bayley, 2021; Rientes & Nolan, 2014; Sauer & Ellis, 2019; Strobl & Baten, 2021; Tracy-Ventura et al., 2016). Social network analysis (SNA) investigates the relationships between individuals ("egos") and their contacts ("alters") and permits the quantification of different aspects of social interaction and communication—such as the structure (topology) of the social network, the strength of the ties between the members, and the direction of the connection—with the goal of furthering understanding of how these relationships influence behaviors and processes such as L2 use and progress. However, little remains known about how communication *among* students on site affects their TL development, how the variables studied change over time, or which factors affect students' social interaction patterns in the first place.

Measuring Social Interactions and Networks

Exposure to and production in the TL during SA have typically been gauged with questionnaires such as various adaptations of the Social Network Strength Scale (SNSS; Milroy & Milroy, 1978), Language Contact Profile (LCP; Freed et al., 2004), Study Abroad Social Interaction Questionnaire (SASIQ; Dewey et al., 2012), Language Engagement Questionnaire (LEQ; R. Mitchell et al., 2017), Social Networking Questionnaire (R. Mitchell et al., 2017), or Daily Linguistic Questionnaire (DLQ; García-Amaya, 2017). Not all of these have succeeded in showing a relationship between the outcome of the measure and L2 proficiency gains (e.g., for a failure to do so by the LCP, see Isabelli-García, 2010; Issa et al., 2020; or Magnan & Back, 2007). In addition, with a few

notable recent exceptions (Arndt et al., 2023b; Baten, 2020; García-Amaya, 2017; McManus, 2019), most of the existing research has focused on students' ties with members of the host community (Isabelli-García, 2006; Campbell, 2011; Baker-Smemoe et al., 2014; Trentman, 2015; R. Mitchell et al., 2017) and on use of the TL, but rarely of both the TL and the first language (L1)—and of other languages, where applicable. Primarily representing the egocentric network approach collecting reports from focal participants only (with the exception of Hasegawa, 2019, and Paradowski et al., 2021a, 2021b, 2022), studies have also tended to concentrate on either single, composite network indices, or more global network measures such as size (number of participants), density (average size of groups in the network), intensity (emotional proximity), or dispersion (number of different social circles the ego is a member of), and have tended to look at only a handful of variables at a time (but see, e.g., Gautier, 2019; Gautier & Chevrot, 2015; Hasegawa, 2019, for more comprehensive approaches).

Longitudinal Social Network Analysis

Until relatively recently, longitudinal SLA studies with multiple measurement points were rare (Barkaoui, 2014; Ellis, 2021, p. 196; Ortega & Byrnes, 2008, pp. 3–4; Tragant & Muñoz, 2023, p. 5). In an article in a volume of the *Annual Review of Applied Linguistics* surveying contemporaneous research trends and future directions, Ortega and Iberri-Shea (2005) noted that “the bulk of disciplinary discussions within the field favors a cross-sectional view of language learning and, as a consequence, discussions about longitudinal research are scarce” (p. 26). They stressed the value of examining development over time and highlighted the potential of multiwave data collection to capture such development. The research community's steps in that direction have been picking up, but longitudinal studies still remain relatively infrequent.

Early SA research was likewise largely built on one-time data collection, but, as early as Barbara Freed's seminal 1995 volume, scholars underscored the need to go beyond cross-sectional models and capture development over time. More recently, the need for a longitudinal bird's-eye view of the development of sojourners' social interactions has been emphasized by Gautier (2019) and Gautier and Chevrot (2015, p. 181). Recent research (e.g., Strawbridge, 2023a) has also called for investigations of how different experiential components in SA affect sojourners' social network and, consequently, language development.

In response to these calls to channel more attention towards longitudinal investigations, the SA field has witnessed the emergence of studies examining

interactions between learners and TL speakers over time (e.g., Kinginger & Carnine, 2019; Shively, 2016), albeit often small in number and mostly with (multiple) case-study designs. For instance, a qualitative investigation by Isabelli-García (2006) of 4 U.S. sojourners in Argentina revealed how extralinguistic variables such as differences in motivation and attitude can affect social interaction in the host culture and culminate in “minimally extended” (p. 232) social networks with native speakers. In the same context but with a larger cohort (23 students), Pozzi (2021) reported that a global network score reflecting quantity and quality of interaction with members of the host community was the strongest predictor of the use of the dialectal informal second-person singular verb form *vos*, and that the strongest social networks with native speakers were created by the most advanced learners. Similar findings demonstrating that social networks with native speakers are statistically significant predictors of the acquisition of L2 sociolinguistic variation patterns, this time in phonology, were reported in Kennedy Terry’s (2017, 2022b) investigations of 17 U.S. students enrolled in a SA program in France.

Serrano, Tragant, and Llanes (2012) collected oral and written production data over three time points from 14 Spanish learners of English in the UK who were part of the European Union’s Erasmus student exchange program. Correlating progress with the number of L1- and L2-speaking roommates, they expectedly found that those living only with English speakers improved their lexical richness more than students who spent most of their time with a Spanish student, which hints at the beneficial role of the affordances for L2 interaction and practice. Strawbridge (2023a) collected egocentric network data for 28 U.S. undergraduate students during a one-semester stay in Spain, at three time points, and found that their social network patterns, which for most students emerged in the first weeks of the semester, remained stable to the end of the sojourn. This mirrors a similar finding by McManus (2019), where the social networks of 29 British undergraduates spending an academic year in France likewise failed to increase over time. In contrast, Gautier (2019), again in an L2 French context but with her 29 students (American and Chinese nationals) coming from U.S. universities, arrived at a different finding: At the beginning of their sojourn, the students mostly formed “dense” networks where most knew one another, almost all their contacts were conationals, and they spent most of their time speaking English; but their networks subsequently evolved into more “extended,” sparser ones with more balanced use of English and French towards the end of the 9-month stay.

Recent approaches such as the experience sampling method (ESM; Arndt et al., 2023a, 2023b) permit large-scale investigations of learners’ language

use at several time points per day, but—because of the intensity of the data collection—the recommended time needed to respond to the questions is usually limited to a few minutes and thus does not easily permit capturing students' networks. In the current study, we take a middle-of-the-road multiwave approach, with students completing detailed questionnaires on their language use (for the L1, TL, and other languages) and social interaction at three equidistant time points during a 3-month sojourn.

The Current Study

In order to connect with conceptual considerations identified in the literature, the current study addresses the following complementary research questions:

1. Which of the social, linguistic, and other variables considered are most influential for both subjectively and objectively measured progress in the TL?
2. Are there observable changes in the patterns of social interaction among the students during the 3 months of the sojourn?
3. Are any differences attributable to participants' gender? Do female and male participants intermingle, or do they tend to stick to same-gender circles?²

Methods and Measures

Social Network Analysis: Theoretical Apparatus

In social network analysis and visualization, each agent—in our case, student—is represented as a node (or, in the network nomenclature, *vertex*), with links (also called edges) signaling relations between these. The focal agent of interest at any given point is often referred to as the *ego*, and their interactants (friends, contacts, etc.) as *alters*. In so-called *egocentric* networks, which reconstruct information based solely on the report of the focal individual, the ego is often not even represented, for the sake of readability. In contrast, *whole* or *complete* sociograms, as in the approach exemplified in this study, rely on information from all or almost all of the participants involved (Sharma & Dodsworth, 2020).

Relations can be directed, indicating that, for example, Student 8 reports that he talked to Student 14 (as shown in Figure 1), as well as weighted, with higher weight meaning that the student perceives their communication with an alter as longer and/or more frequent (or “durable” in the nomenclature of some extant SA literature) than their communication with another (e.g., that with Student 1, for Student 8 in Figure 1). The social network recreated in this

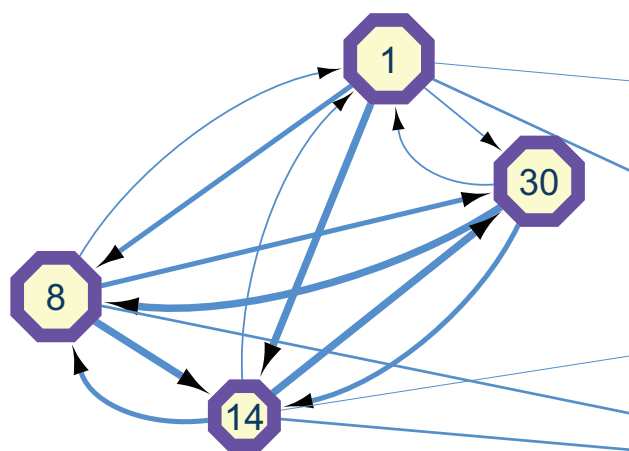


Figure 1 Visualization of social network interactions (detail of Figure 18 later on). Arrows show each member's up to five top declared contacts/interlocutors (some nodes outside this zoomed-in view), with arrow thickness reflecting reported frequency/intensity of communication.

study is temporal (Bródka et al., 2013) and consists of three snapshots (also known as timeframes), representing the state of the network after the 1st, 2nd, and 3rd month of the sojourn, respectively.

In the analyses, we have calculated several metrics that help to characterize each student based on their position in the social network (Newman, 2018). The established network measures considered here include (a) five classic centrality metrics: degree, in-degree, out-degree, closeness, and betweenness; and (b) three newer indices: PageRank, voterank, and clustering coefficient. These are all explained below.

Centrality means that the node is central or important from a particular point of view. *Degree* centrality tells us how many ties a focal student has; that is, with how many others (alters) they have been able to build a relation and stay in contact. Degree can be further divided into two dimensions: *out-degree*, reflecting how many alters the student reports talking to, and *in-degree*, reflecting how many others indicated that student as their interlocutor. In terms of language acquisition, these two metrics might be treated as imperfect proxies of the number of people exposed to a given student's output, and the number of people from whom the student receives input, respectively (some symmetry in the communication notwithstanding). High in-degree, even when only TL communication is considered, may be detrimental to overall progress (Paradowski et al., 2022), probably because it takes place at the expense of output, which latter may act as the primary trigger of interlanguage

restructuring, though high in-degree may aid the acquisition of vocabulary (Paradowski et al., 2021a, 2021b).

Closeness centrality eponymously reflects how “close” the student is to all the other students—or central in the network (Cartier van Dissel et al., in press)—and thus, for example, how fast they can get information, knowledge, or skills from and to the other students. In a SA context, high closeness with other foreign peers in the general communication network may negatively correlate with progress (Paradowski et al., 2022), because it signifies being either close to everyone often at a superficial level, or relatively solitary, positioned between smaller groups but without belonging to them. Also, students with high centrality in the general communication network may more often be resorting to a lingua franca than the TL, limiting their exposure to and practice in the latter.

Betweenness indicates the proportion of all the shortest paths in the network that go through the student. People with high betweenness centrality act as “bridges” strategically connecting others and are key in controlling and passing information and knowledge between network communities as well as maintaining relations between them. High betweenness in peer L2 learner networks has been associated with depressed self-perceived progress in vocabulary and pronunciation, probably due to relatively higher exposure to nonnative models, and possibly resulting from coidentification with conationals (Paradowski et al., 2021a, 2021b).

The *clustering coefficient* (Watts & Strogatz, 1998) serves as indication of the node’s social embeddedness, or how well connected their neighbors are. It does so by calculating the proportion of the existing interpersonal connections among the neighbors to the number of all the theoretically possible connections if the neighbors formed a complete graph (clique). It is thus similar to another measure of cohesion found in the SA literature, network density (Borgatti et al., 2024; Wasserman & Faust, 1994), the difference being that the latter expresses the ratio of actual to total possible ties *including* the ego. Nodes with a high clustering coefficient are usually in the center of local communities (groups), whereas a low clustering coefficient, especially in conjunction with high betweenness, indicates that the node is a bridge connecting two or more separate communities in the network. The clustering coefficient may then be seen as a different, inverse way of looking at dispersion, a measure used in prior SA studies to reflect the average number of groups an individual was a member of. The clustering coefficient was found to negatively correlate with L2 progress in a group of Erasmus students in Germany (Paradowski et al., 2022).

A yet different measure commonly used to determine the relative influence or importance of a node in *undirected* networks is *eigenvector centrality*. Unlike cruder measures such as degree, which only counts the number of connections a node has, eigenvector centrality also takes into account the centrality of the nodes to which the focal agent is connected (Landau, 1895). Thus a student will have a high eigenvector centrality score if they are connected to other students who are highly central themselves. A variant of eigenvector centrality better suited to *directed* networks such as ours is *PageRank* (Brin & Page, 1998), best known from Google's search result positioning algorithm. Using this metric, a student is seen as more important or popular if they are perceived as a friend by other important or popular students. This measure thus differs from in-degree in that, instead of taking into account only the student's (immediate) neighbors, the entire network is considered.

Finally, *voterank* ranks nodes based on their potential to influence other individuals in the network while avoiding the problem of influential nodes clustering together (Zhang et al., 2016). This measure finds application in the influence maximization problem, where one looks for the smallest set of the most influential nodes who can help sway the whole network. From a language learning perspective, people with high voterank can be perceived by teachers as the best contact points to disseminate information to the class or serve as role models of beneficial behaviors.

The Whole-Network Approach

Addressing the lacunae in most existing SA SLA research, we go beyond a focus on interactions with speakers of the TL, reconstruction of ego-networks only, and undirected and unweighted networks, in order to investigate the patterns, and impact on progress, of mutual interactions with peers from the same alma mater taking part in the SA program. Such an approach has the advantage of permitting analyses of the structural position of each student in the complete network and, instead of global network parameters, of allowing the calculation of mutually dependent indicators (Cartier van Dissel et al., in press) such as interaction frequency as well as a range of centrality metrics.

A frequent assumption regarding L2 development during SA concerns the role of TL contact with locals as (one of) the strongest predictor(s) of language gains. Although we also asked about the students' interactions outside of their cohort, our reconstruction of their social networks focused on peers from their home institution. This was motivated by three main considerations:

- During intensive programs such as the one investigated here, interactions with fellow foreign students constitute a large (if not the major) part of the sojourners' contacts. In our case, this proportion ranged from 59% during the 1st month of the sojourn through 58% during the 2nd month to 56.6% in the last month, while the amount of self-reported *TL* communication (whether in face-to-face interactions only, or also including emails, social media posts, texting, and voice calls) failed to affect progress.
- The literature corroborates the challenges of accessing and establishing meaningful interactions with members of the local community (other than household members in homestays), especially in the case of short-term programs (e.g., Allen, 2010; Goldoni, 2013; Hasegawa, 2019; Kubota, 2016; McManus, 2019; Trentman, 2013a, 2013b). A further relevant challenge discussed in the literature is lack of prior preparation in local Arabic dialects (Shiri, 2013).
- Given the reciprocal network approach, focus on course participants allows the collection of responses not only from each focal student (ego), but also from their contacts (alters). Although study-abroad research focusing on *TL* contacts only and utilizing isolated egocentric networks (without information on what the interactions look like from the perspective of the alters) may be insightful in situations where sojourners spend relatively more time communicating outside their fellow learner network than within,³ the dominance of interactions with peers from the home university in our study validates the rationale behind our focus on the peer learner network, alongside the affordance of reconstructing the whole social graph.

The data were collected by using a specially developed PEERLANG questionnaire adopted from work by Paradowski et al. (2021a, 2021b, 2022; the materials, data, and code used for the analyses can be accessed at <https://www.iris-database.org/details/aRtbx-9zI2E> and <https://osf.io/3gcyw/>), administered in situ after each month of the sojourn. This asked the students about a range of background and attitudinal/affective measures, their competence in other languages,⁴ their language use and communication with each other classmate from their home institution (aka the name interpreter approach) as well as with other groups of interlocutors, their subjective assessment of how well their group were internally integrated, and their self-perceived progress, both overall and across specific skills/competencies, during the month preceding each current survey administration. Additional insights were gained in discussion with

the program coordinator who had been with the participants on site throughout its duration and thus had in-depth insight into, and first-hand information on, their daily lives and progress. Objective progress (global) was measured by subtracting pre-sojourn oral proficiency scores from post-sojourn scores. The scores were drawn from the ACTFL Oral Proficiency Interview (OPI; American Council on the Teaching of Foreign Languages). The protocol had obtained Institutional Review Board approval #53/2020 from the Rector's Committee for Human Research Ethics at the University of Warsaw.

Statistical Analyses

To obtain the most reliable and stable results in the inferential analyses, we used the mean of the variables of interest from across all three months. Where interesting significant patterns emerged at only some of the time points, we comment accordingly. To identify variables contributing to objectively measured progress, calculated as increase in the OPI score, we ran a multiple regression analysis. For the computational network analyses, we used Python libraries Network Diffusion⁵ (Czuba & Bródka, 2022), NetworkX,⁶ and Matplotlib,⁷ which allow for temporal network analyses (using longitudinal data). To gauge relationships between network metrics, self-perceived progress across language subskills/competencies, and other individual variables, due to the occasional lack of a clear directionality, correlation coefficients were calculated following checks for normality. Normality was inspected by calculating *z*-score values for skewness, with the limit set at $|1.96|$ (Kim, 2013; Mayers, 2013, pp. 52–53). For correlations involving skewed data, nonparametric Spearman's ρ is reported in place of Pearson's r and signaled accordingly. To measure the impact of gender, we calculated independent-samples *t*-test values; homogeneity of variance was measured with Levene's test and was not violated.

A check for outliers in the predictors of the regression model was carried out, with inspection of Cook's distance (all values contained within the 0–0.168 range, thus none above 0.5), Mahalanobis distance (one participant with a value of 15.228, marginally acceptable for five predictors at χ^2 significance level of .009), and standardized residuals. One participant's residual was more than two standard deviations away from the mean (-2.281), but this constituted 3.8% (thus less than 5%) of all cases, and did not influence the results (most extreme standardized DF beta value of $-.9313$, below $|1|$). Quantile–quantile plots and boxplots were also inspected and likewise did not identify significant outliers. Normality of residuals was ascertained by inspecting the PP plot. Whereas White's test for heteroscedasticity ($\chi^2_{(20)} = 21.085$,

$p = .392$) returned a negative, the Breusch–Pagan test, as well as its modified variant, signaled a violation of the assumption of homoscedasticity of residuals ($\chi^2_{(1)} = 4.682$, $p = .030$, and $\chi^2_{(1)} = 3.825$, $p = .050$, respectively), confirming the eyeball test. We accordingly ran a regression with heteroscedasticity-consistent (“robust”) standard errors (Hayes & Cai, 2007), using the Cribari-Neto covariance matrix estimator (HC4; Cribari-Neto, 2004). Dynamic changes in the data over the 3 months are visualized in violin plot panels (Figures 3–17) generated in Python with the seaborn package (Waskom, 2021), allowing for easy comparison of the trends. Correlations were calculated in SPSS 29, and regression in R using the sandwich and rempsyc libraries; dominance analysis was carried out using code from Mizumoto (2023).

Network Diagramming

With the complete peer network being highly interconnected and occluding individual differences with respect to network positioning, we instead adopted an approach focusing on interactions recreated from reports of each student indicating up to five of their most frequent contacts (name generator approach). This yielded a much more transparent picture, with much clearer heterogeneity in terms of the position of the students in the peer network. The sociocentric network visualizations were performed with Cytoscape⁸ (Shannon et al., 2003), selecting the Compound Spring Embedder (Dognuz et al., 2009) as the visualization layout.

Participants and the Study Abroad Program

Study Abroad Program Characteristics

The participants were 41 learners of Arabic enrolled in an intensive semester-long SA “island” program, in which the sojourners attend dedicated classes, but do not take part in courses with local students, in Amman, Jordan. The program was organized by a large private university in the USA and was based in a private language institute led by a faculty member from the home institution. Students regularly hear about the Jordan SA program from the beginning of their Arabic study. The curriculum involves an integrated approach where Levantine Arabic is used for most of the oral communication. The participants’ daily routine (five days a week) consisted of several different elements:

1. There was a 2-hour daily content class that consisted of discussions in Arabic of societal issues of importance in Jordan, culminating in weekly in-class presentations by the students. Class size ranged from 4–10 students, with more advanced students in the larger classes.

2. There was a 75-min-per-day course focused mostly on the language of news media and occasional literary or religious readings (e.g., from the Qur'an, Hadith, or Bible) that students read or listened to on their own in preparation for class. This regime of daily intensive and extensive readings and listening activities is a key component of the program that enables students to acquire facility in speaking and writing about a limited number of topics of particular interest to locals, thereby facilitating conversations on issues well beyond the everyday survival and socializing needs of the participants. Students would break into three or four small groups for the last 20–30 min of this class. The more advanced students typically met with an Arab instructor and discussed the content of the reading or listening assignments in Arabic, whereas the rest focused on linguistic details, almost exclusively in English. As students increased in proficiency, native speakers of Arabic were increasingly invited to take more of the class time to talk about matters connected with the readings and holidays, such as the Birthday of the Prophet Muhammad. The program director and two teaching assistants, who had participated in this university's SA program the preceding year and served as role models, provided strategy training for language learning and for cultural adaptation both in this class throughout the week and in weekly individual interviews with the students designed to spot-check their reading of assigned intensive articles and discuss their ongoing experiences.
3. Approximately 20 min a week were used for a group processing experience where students talked about cross-cultural and language-learning challenges and discussed ways of understanding and coping with these challenges with one another and with program staff.
4. Students were required to spend 2 hours a day outside of class speaking Arabic with native speakers. They partially satisfied this by meeting with trained Jordanian tutors for 30 min each day.
5. Students also had three 30-min speaking presentation appointments with native tutors each week and two 15-min appointments to discuss their own Arabic writing with these tutors.

For more details on the curriculum and history of the program, see Bird and Belnap (2018).

Students are drawn to this specific program for its reputation as a truly intensive experience that results in significant gains in proficiency for most participants. It is required of Middle East Studies/Arabic (MESA) majors and of

Arabic Language majors, but also sees participation by students from a variety of other majors, such as Linguistics, Political Science, Finance, and Exercise Physiology.

Participant Profile

The participants were a complete cohort of 41 learners of Arabic (17 female, 24 male; aged 19–29 years) enrolled in the above-described semester-long program. All were native speakers of English, with the exception of one Chinese national. All were students at the organizing university, representing a total of 23 majors, and had completed four semesters of Arabic instruction (50 min per day, five days per week) prior to beginning the SA. Preprogram OPI scores ranged between Intermediate-Low and Advanced-Low.⁹ The average score was Intermediate-Mid, roughly equivalent to A2/B1 according to the descriptors in the *Common European Framework of Reference for Languages* (CEFR; Council of Europe, 2020), but nearly half of the students (46%) scored considerably better than that, at least Intermediate-High. Nearly all of the participants had lived in another country before this SA experience and had become proficient in a L2. The influences motivating TL use were compliance with the grade-bearing requirement of speaking with L1 Arabic speakers for 2 hours each day, conversing with new friends in Amman, and—for MESA majors—continuing study the following year at a higher level in Morocco.

The students live in apartments with other program participants or with a spouse. The students in the cohort at hand had been in classes together and had become good friends; a number requested that they be roommates in Jordan. Single females were housed in three apartments, and single males in four. Three participants (Nodes 10, 37, and 39 in the social graphs in Figures 18–20) lived with their spouses, who were not participating in the program. In Euclidean distance (as the crow flies), they had roughly no more than half a mile, or 800 m, to their furthest peer. Three (Nodes 2, 6, and 33) lived with their families on the other side of the highway, extending the walking distance to 1 mile (Figure 2). For safety reasons, the students were asked to be in pairs when not in the vicinity of their apartment or the nearby language institute, and after dark in triads that had to include at least one male, given the tendency for female students to experience sexual harassment. Two of the participants were married to each other; three other couples were dating. Six other students, five of them men, had a spouse who spoke little or no Arabic.

Results

We present the results for each of the three research questions in turn.

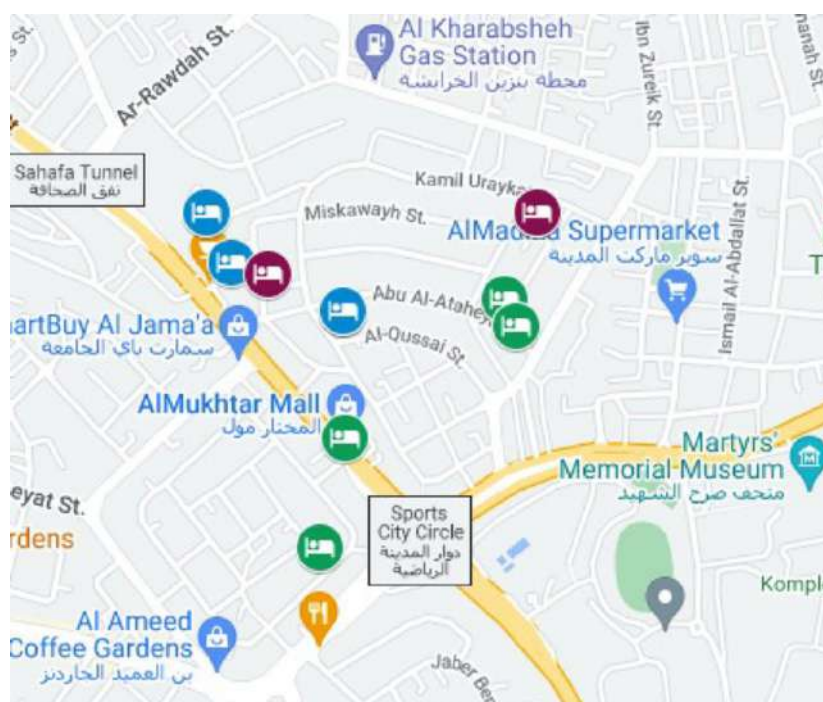


Figure 2 Participants' accommodation locations. Circular blue markers indicate male students, purple ones female students, and green ones families. One location may conceal more than one apartment. Map data ©2024 Mapa GISrael, Google; reproduced under fair use principles.

Influences on Progress

The first research question asked which of the variables considered were most influential for both subjectively and objectively measured progress.

A few of the network measures significantly correlated with self-reported progress in selected skills/competencies, although in some cases this relationship was only observed during some of the months. Thus, betweenness centrality, in simple terms measuring the extent to which a node strategically connects others, negatively correlated with the positive perception of own proficiency relative to that of classmates, $r_s = -.391$, 95% CI $[-.641, -.067]$, $p = .017$,¹⁰ but positively with self-reported improvement in reading, $r_s = .453$, 95% CI $[.142, .683]$, $p = .005$ (primarily affecting the last month of the sojourn, $r_s = .481$, $p = .008$), and with subjectively perceived gains in grammar, albeit only in the middle month of the stay ($r_s = .372$, $p = .047$). Voterank, the measure of community influence, seemed to negatively predict progress in writing, $r_s = -.402$, 95% CI $[-.674, -.019]$, $p = .038$, even though students with a higher level of voterank reported spending more time

learning Arabic outside of class in the middle month of their stay in Amman ($r_s = .487, p = .029$). That month also saw a positive relationship between the clustering coefficient and self-perceived progress in culture ($r_s = .408, p = .013$) as well as listening ($r_s = .418, p = .011$).

Objectively measured progress was operationalized as the gain from the pre- to the postsojourn OPI score. Pre- to post-OPI negatively correlated with predeparture proficiency, $r = -.485$, 95% CI $[-.695, -.184]$, $p = .002$, and positively with final scores, $r = .421$, 95% CI $[.107, .652]$, $p = .009$. Other influential variables were use of Arabic outside of class, $r = .371$, 95% CI $[.048, .617]$, $p = .024$ (primarily influenced by the last month of the sojourn, $r = .517, p = .006$); perception of group integration (negatively influential), $r_s = -.418$, 95% CI $[-.659, -.099]$, $p = .010$ (here most strongly influenced by the middle month of the sojourn, $r_s = -.538, p = .001$); and (also negatively influential) the proportion of interaction with alma mater classmates, $r_s = -.372$, 95% CI $[-.618, -.049]$, $p = .023$ (primarily influenced by the 1st month of the sojourn, $r_s = -.460, p = .005$). L2 gains may also have been helped by the time spent learning Arabic outside of class during the final month of the sojourn ($r_s = .368, p = .049$) and by betweenness centrality in the middle of the stay ($r_s = .345, p = .039$).

In the next step, in order to ascertain the relative impact of the independent variables on objectively measured progress, we ran a regression with heteroscedasticity-consistent (“robust”) standard errors. After removing spurious variables as well as predictors resulting in multicollinearity, we ended up with a model including five predictors that together explained nearly half of the variance in objective L2 progress, $F_{(5,21)} = 5.485$, adj. $R^2 = .463$, $p = .002$ (see Table 1). The significant predictors are presojourn proficiency in Arabic as measured by the OPI (negative; large effect size), perceived integration of the U.S. student group (negative; medium to large effect, depending on the cut-off recommendation adopted; see Foster, 2018, p. 209 vs Richardson, 2011), in-degree (the number of alters speaking to the student; small to medium effect), and degree of multilingualism (competence in other languages; medium to large effect).

Changes in Social Interaction Patterns

The second research question asked whether there were observable changes in the patterns of social interactions among the students during the 3 months of the sojourn.

In terms of longitudinal development, we observe a visible dip in use of the TL outside of class (Figure 3), time spent learning the TL outside of class

Table 1 Predictors of objectively measured progress, in order of impact

Predictor	<i>b</i>	95% CI	<i>SE</i>	β	95% CI	<i>t</i>	<i>p</i>	η_p^2	VIF	Dominance weight (%)
(intercept)	.496	[.328, .664]	.081	.176	[−.147, .498]	6.141	< .001			
Presojourn OPI	−.425	[−.738, −.111]	.151	−.477	[−.829, −.124]	−2.814	.010	.267	1.494	.252 (44.49%)
Perception of group integration	−.238	[−.406, −.071]	.081	−.494	[−.840, −.146]	−2.958	.008	.215	1.392	.165 (29.17%)
In-degree (centered)	.023	[.003, .044]	.010	.269	[.034, .504]	2.380	.027	.124	1.051	.067 (11.81%)
Competence in other languages	.010	[.001, .019]	.004	.381	[.029, .733]	2.251	.035	.194	1.176	.055 (9.66%)
Voterank (centered)	−.002	[−.007, .003]	.002	−.136	[−.435, .162]	−0.948	.354	.025	1.414	.028 (4.87%)

Note. Regression with heteroscedasticity-consistent standard errors (Cribari-Neto HC4 covariance matrix estimator). Boldface indicates significant results at $p < 0.5$. *b* – unstandardized regression coefficient; CI – 95% confidence interval; SE – standard error; β – standardized regression coefficient. VIF = variance inflation factor; OPI = Oral Proficiency Interview.

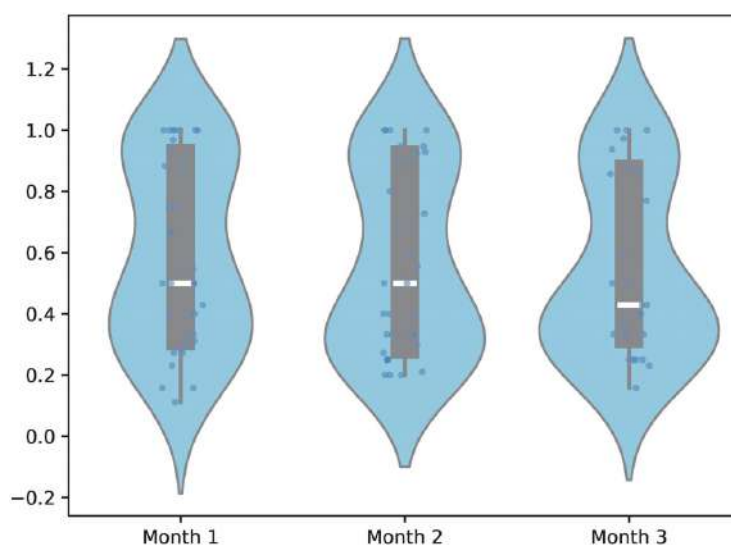


Figure 3 Change in target language use outside of class over the sojourn (1-month intervals), understood as percentage of Arabic use out of total communication. In this and subsequent graphs, the white middle line indicates the median value, the boxplots the two middle quartiles.

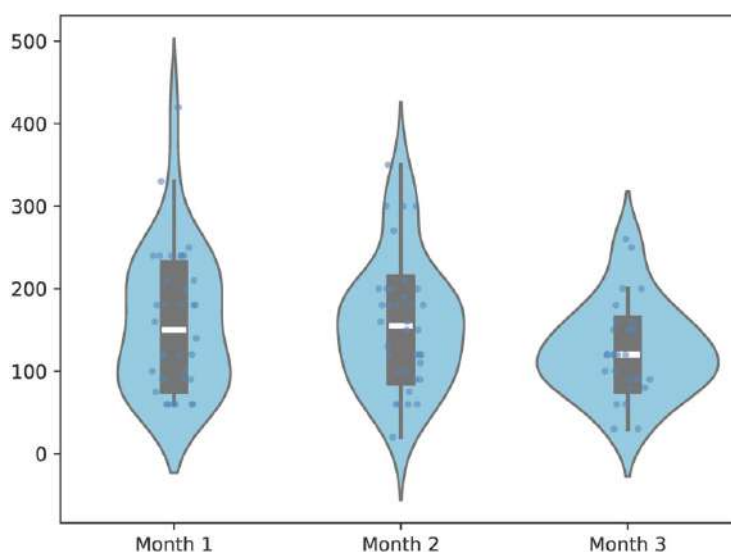


Figure 4 Change in time spent learning the second language outside of class, in minutes per day.

(Figure 4), and percentage of out-of-class interaction with classmates during the last month of the sojourn (Figure 5).

Expectedly, there was a visible trend of perceived integration of the group steadily growing from month to month (Figure 6).

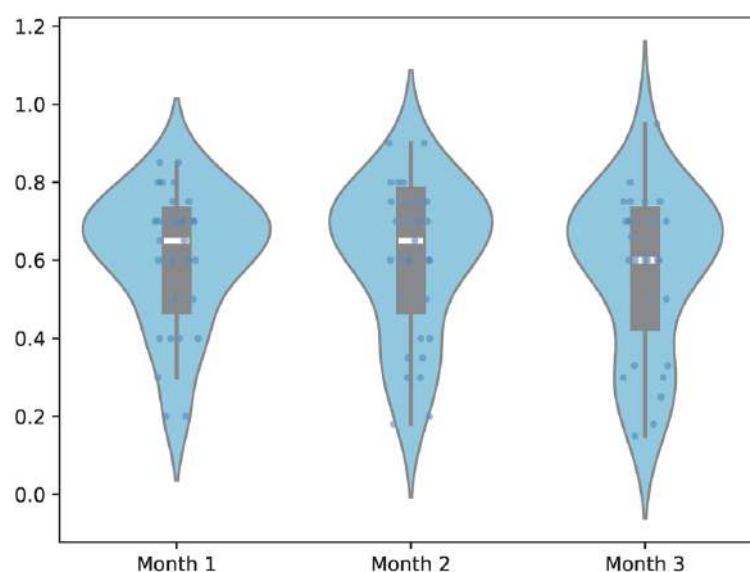


Figure 5 Change in proportion of out-of-class interaction with alma mater classmates out of total communication.

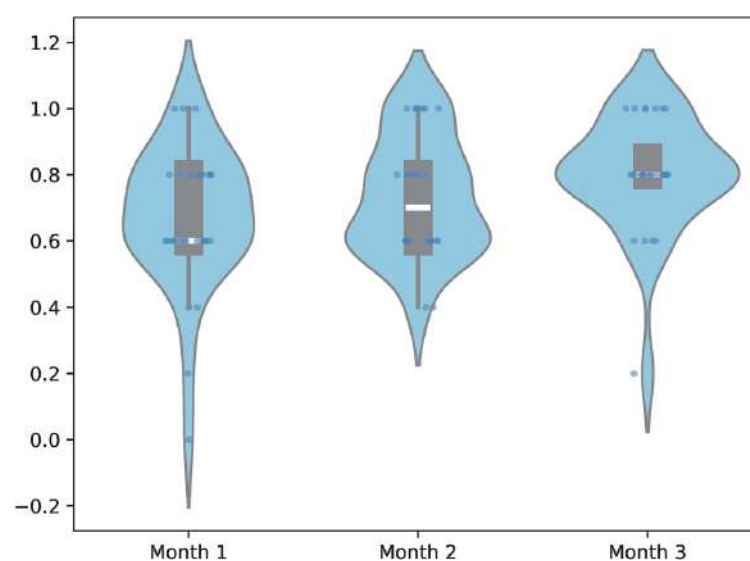


Figure 6 Change in level of perceived integration of the student group.

Self-perceived improvement in reading and vocabulary dropped after the 1st month (Figures 7 and 8), whereas perceived progress in listening and grammar saw an improvement in the 3rd month (Figures 9 and 10).

Most of the centrality metrics saw a decrease over the period of the sojourn. In-degree (the number of alters speaking to the student) and clustering coeffi-

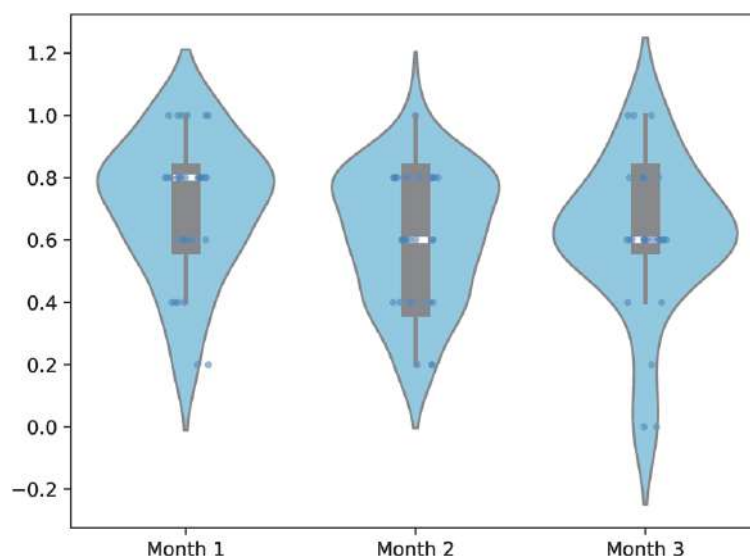


Figure 7 Change in level of perceived progress in reading.

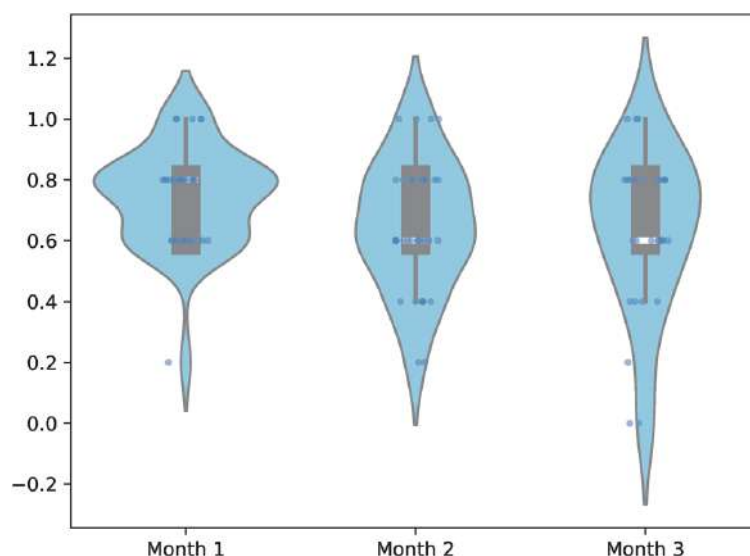


Figure 8 Change in level of perceived progress in vocabulary.

cient dropped after the 1st month (Figures 11 and 12). Out-degree (the number of alters the student talked to) and overall (undirected) degree—number of frequent contacts—decreased in the 3rd month (Figures 13 and 14), as did closeness (Figure 15) and voterank (the measure of optimal influence; Figure 16).

PageRank (exposure to other “popular” students) in turn increased over the last month (Figure 17).

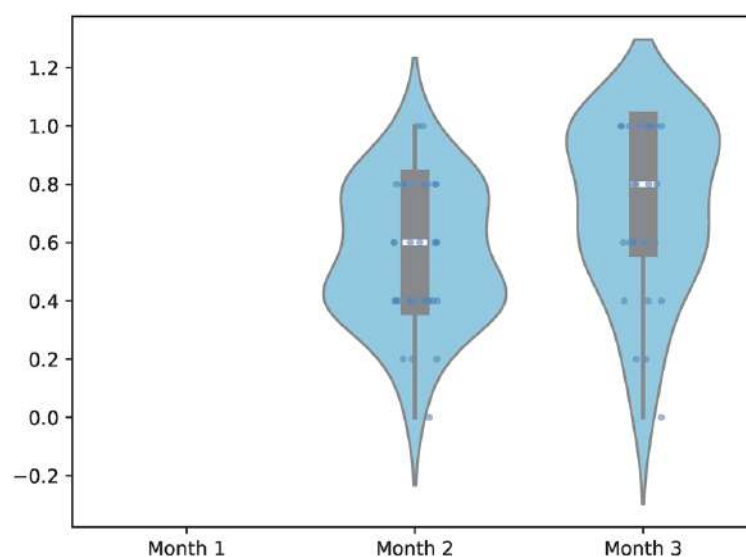


Figure 9 Change in level of perceived progress in listening (1st month missing due to an oversight in the questionnaire).

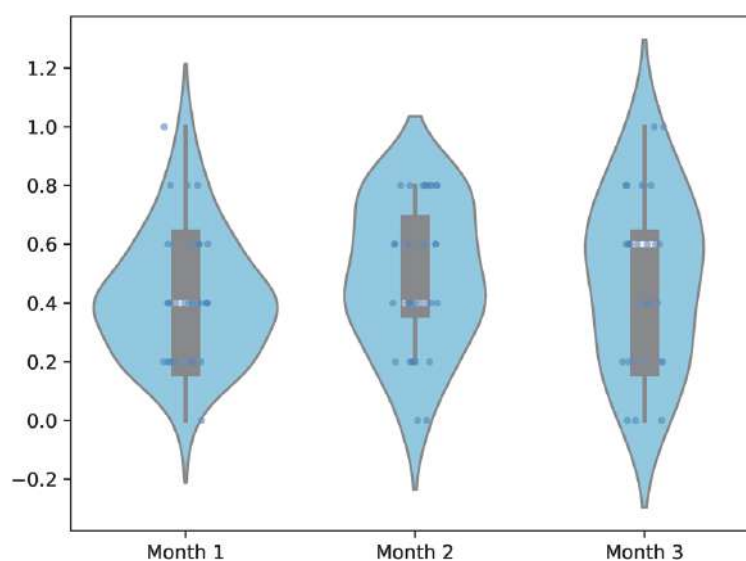


Figure 10 Change in level of perceived progress in grammar.

The Influence of Gender

The third research question asked whether there were differences attributable to participants' gender, and whether female and male participants intermingled or tended to stick to same-gender circles.

Snapshots of the student network taken after each subsequent month reveal relative stability in terms of position in the graph, but with some visible

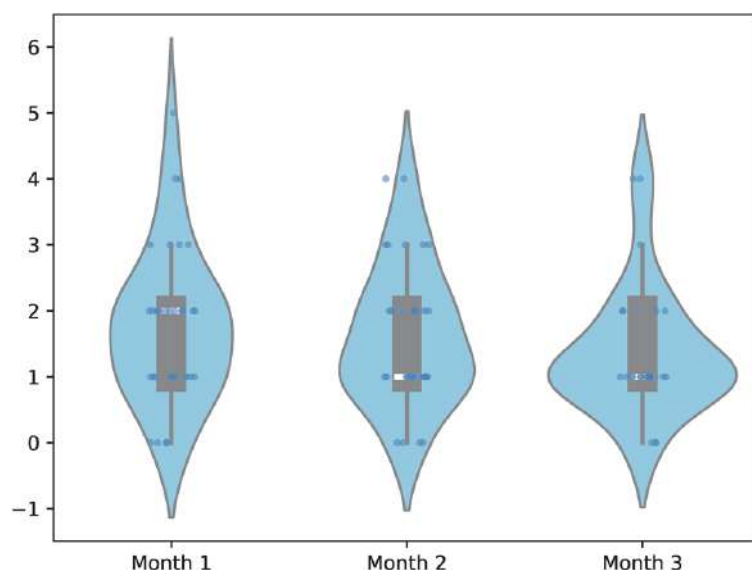


Figure 11 Change in students' in-degree.

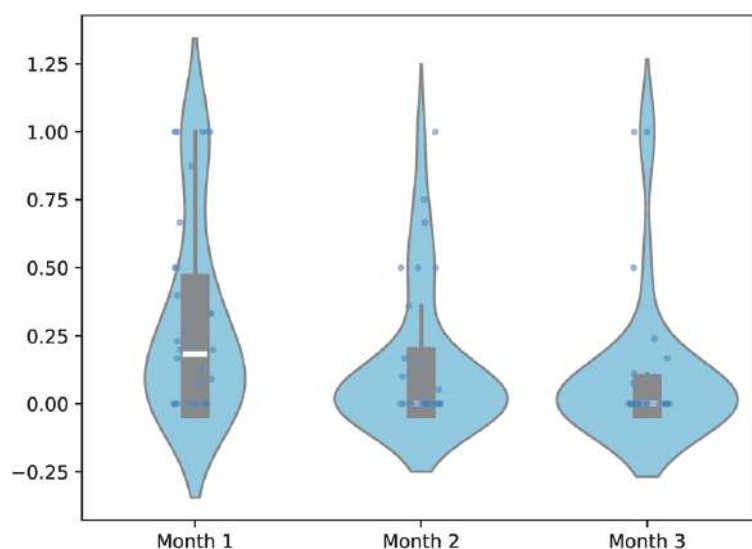


Figure 12 Change in clustering coefficient.

changes in the patterns and strength of interaction (Figures 18, 19 and 20). Immediately striking is the stable tendency of the students to form cliques with peers of the same gender; exceptions are few and far between, are mostly female, and tend to appear on the edges of subcommunities, linking these with neighboring clusters. With time, this gender homophily (associating with similar others) only strengthens.

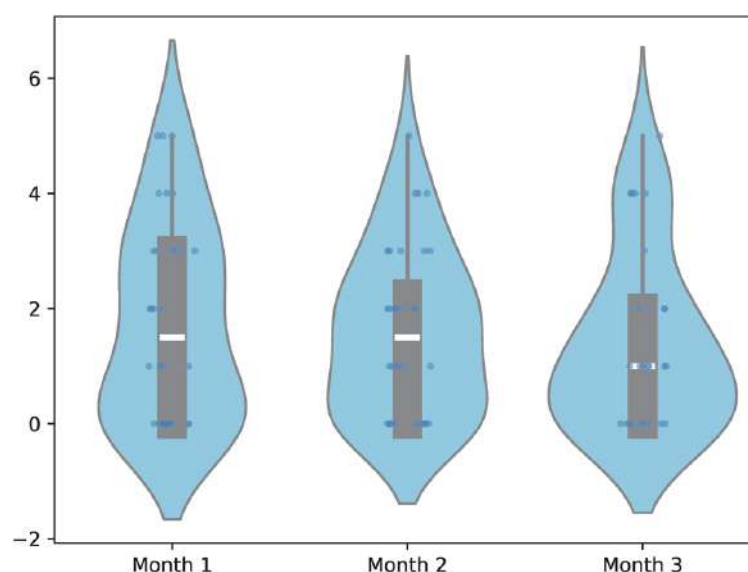


Figure 13 Change in students' out-degree.

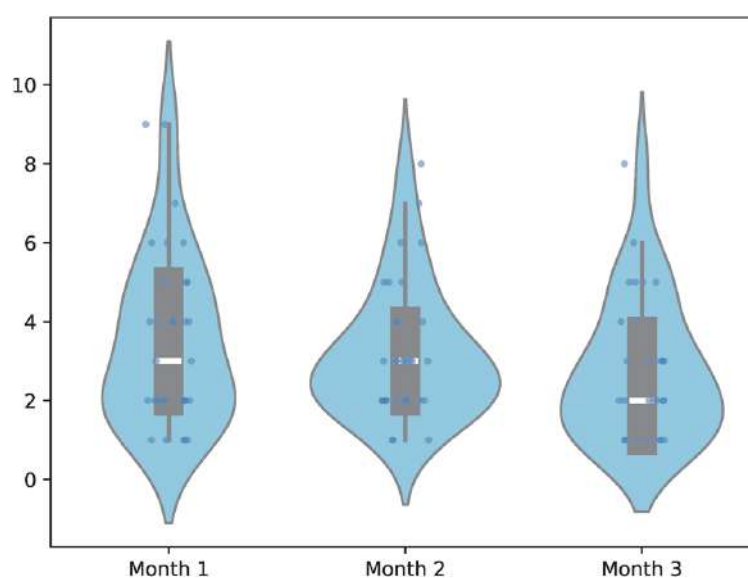


Figure 14 Change in students' degree (undirected).

The female participants also carried out a higher proportion of their communication with their alma mater classmates than did their male colleagues ($t = -2.473$, $p = .018$, $d = -0.780$, 95% CI $[-1.451, -0.100]$; especially during the 1st month of their sojourn), probably in part because the female students also tended to share their accommodation with more peers

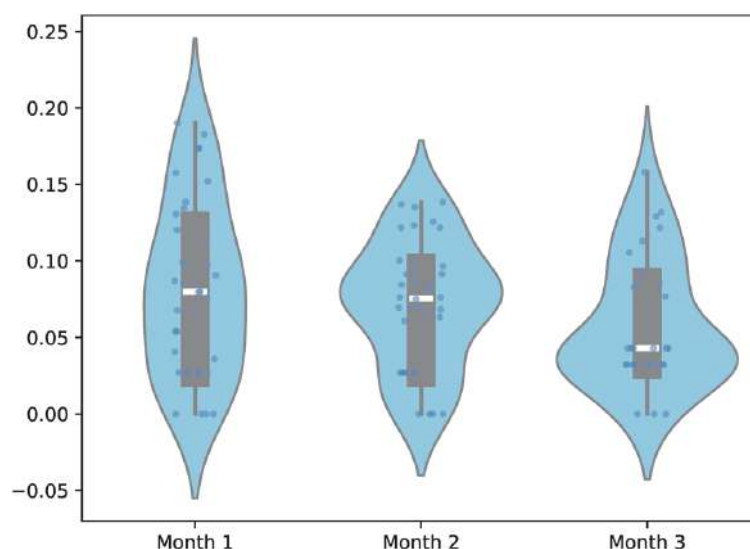


Figure 15 Change in students' closeness centrality.

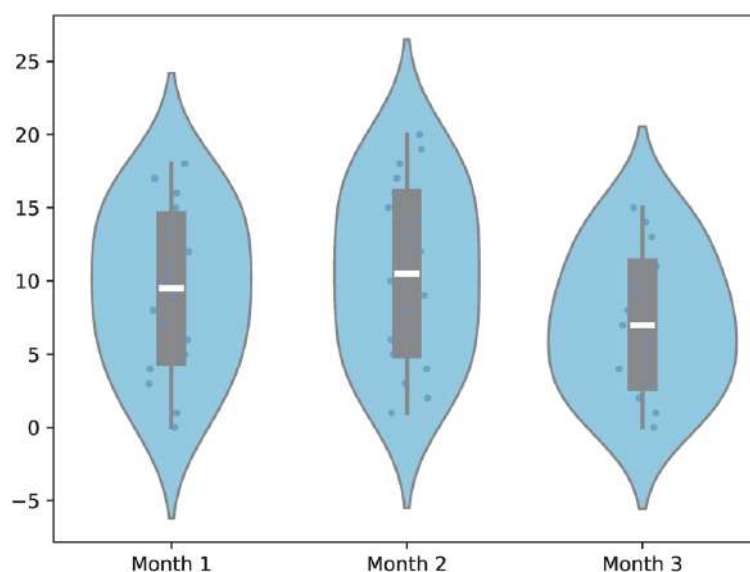


Figure 16 Change in students' voterank.

($t = -2.678$, $p = .014$, $d = -0.955$, 95% CI $[-1.637, -0.262]$). Other significant differences attributable to participants' gender included male participants' both starting out and finishing with relatively higher objectively measured proficiency in Arabic (initial: $t = 2.403$, $p = .022$, $d = 0.797$, 95% CI $[0.116, 1.469]$; finishing: $t = 2.919$, $p = .006$, $d = 0.969$, 95% CI $[0.274, 1.651]$; Figure 21), as well as displaying higher degrees of multilingualism ($t = 2.706$,

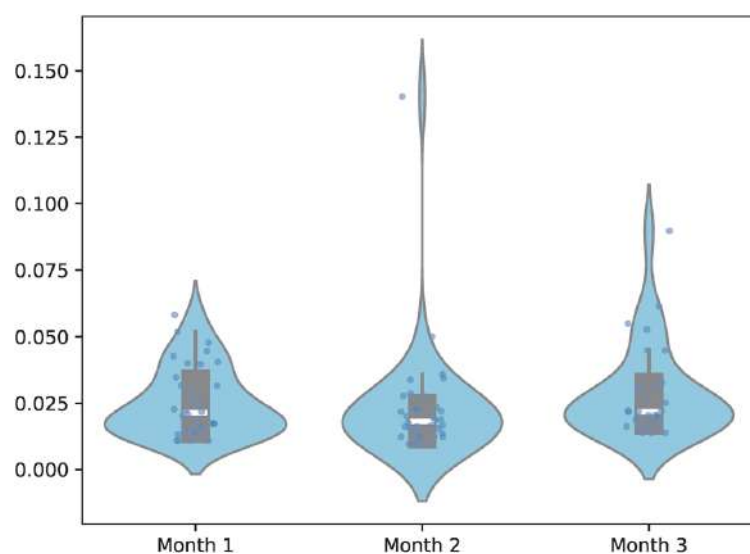


Figure 17 Change in students' PageRank.

$p = .010$, $d = 0.898$, 95% CI [0.209, 1.576]). However, subjectively, it was the female participants who on average felt that their level of proficiency was higher than that of their classmates ($t = -4.738$, $p < .001$, $d = -1.511$, 95% CI [-2.242, -0.763]). As can be seen from the parenthetical Cohen's d values, all these differences show large effect sizes.

Discussion

Social Network Position and Language Gains

The finding of betweenness negatively correlating with perception of one's own proficiency may suggest that having more relations and information on how the other participants are faring permits a more cautious calibration of self-assessment. The positive relationship between this metric and self-reported progress in reading may imply that higher betweenness resulted in more success in dealing with the increased workload that the students encountered, since the volume of required readings had increased from one short news article or literature assignment per week during semesters at the home university to two to four intensive and long, extensive articles per day during the sojourn, which resulted in routine collaboration by the students in the translation and comprehension of the assigned texts. This hypothesis gains additional support from the positive association of betweenness also with both objectively measured holistic progress and subjectively perceived progress in grammar in the middle month of the stay, and from in-degree (the number of

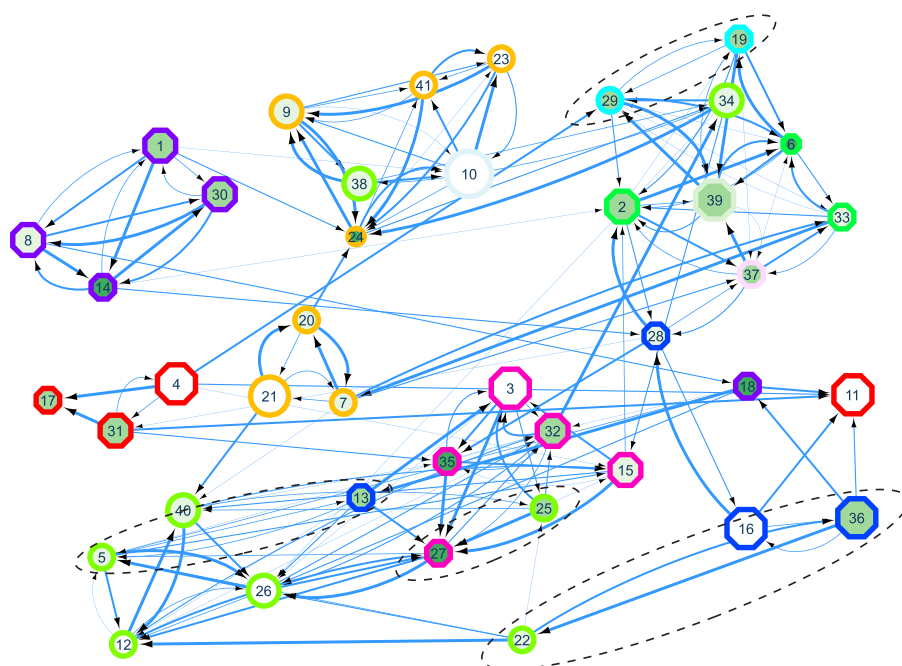


Figure 18 Network of peer interaction among the 41 study abroad sojourners during the 1st month of the program. Circles denote female, octagons male participants. Same border color indicates proximal accommodation (apartments in the same location but not necessarily shared) in the city's topography. Node color intensity reflects presojoir Oral Proficiency Interview score; node size objectively measured postsojoir progress. Arrows show each student's top five declared contacts/interlocutors (upper limit set for transparency), with arrow thickness reflecting reported frequency/intensity of communication. Dashed-line ellipses highlight students who were married to or dating each other at the time of the sojoir. Visualization generated from questionnaire data using Cytoscape (Shannon et al., 2003) with the Compound Spring Embedder layout (Dogru-soz et al., 2009).

alters speaking to the student) being a significant predictor of L2 gains. The learners participating in this SA program, per administrators' request to the host language institute, had specifically asked their local Arabic instructors to not teach grammar but rather focus on speaking. This resulted in students' consulting with one another to clarify grammatical constructions. Each day also included 15 min to exchange dialectal forms acquired the previous day during the TL-speaking time with locals or with the course supervisor from their home university. This mutual formal instruction thus benefited the students' grammar development. Some SA research also suggests that L2 learners abroad regularly find opportunities to interact with others in the TL through members of their own peer networks (Dewey, Ring, et al.,

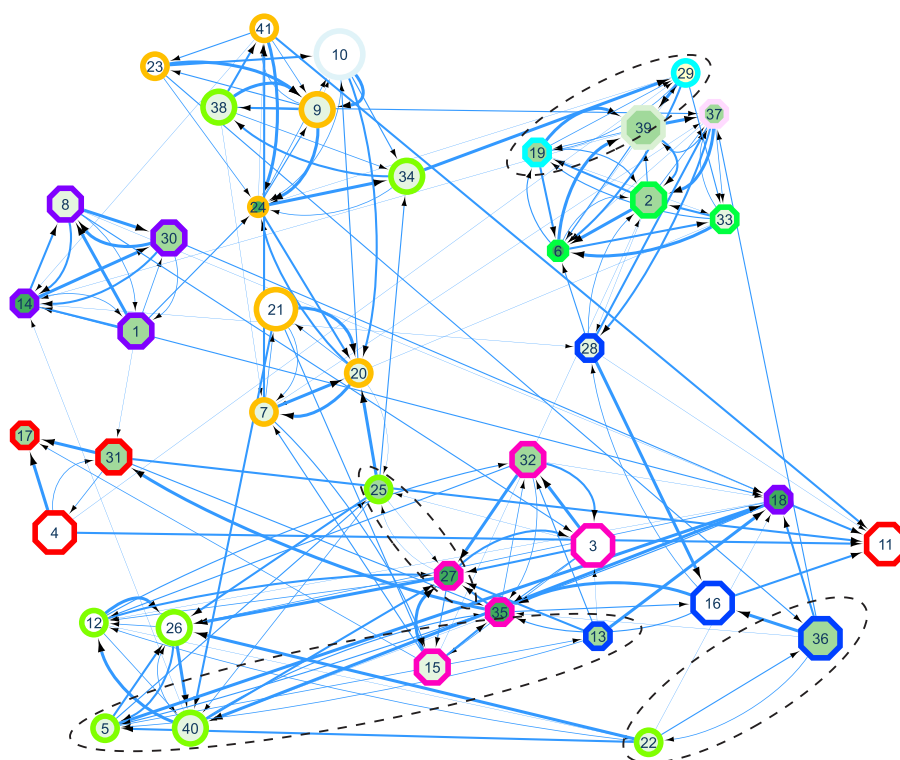


Figure 19 Network of peer interaction among the 41 study abroad sojourners during the 2nd month.

2013; Ring et al., 2013). In other words, they are introduced to local TL speakers by classmates and are connected socially in ways that allow them to use the L2 in authentic contexts. It is therefore natural that such social capital (Bourdieu, 1986) can lead to more L2 use, with both classmates and locals.

In turn, writing saw a negative correlation with voterank. This measure of community influence indicates “leaders” who would be optimal information spreaders and influencers due to their numerous and diverse relations. Centrality measures are first and foremost the result of oral, not written communication, which instead is a predominantly individual activity practiced in the classroom and in home assignments, during times when spoken interaction is *not* taking place. In addition, writing development is not typically cited as a reason for going abroad. Though this SA program was not devoid of practice in this skill, the latter was mostly worked on as an individual exercise. Literature suggests that while a few months abroad might be sufficient for gains in oral performance, a short-term SA context is not especially helpful for

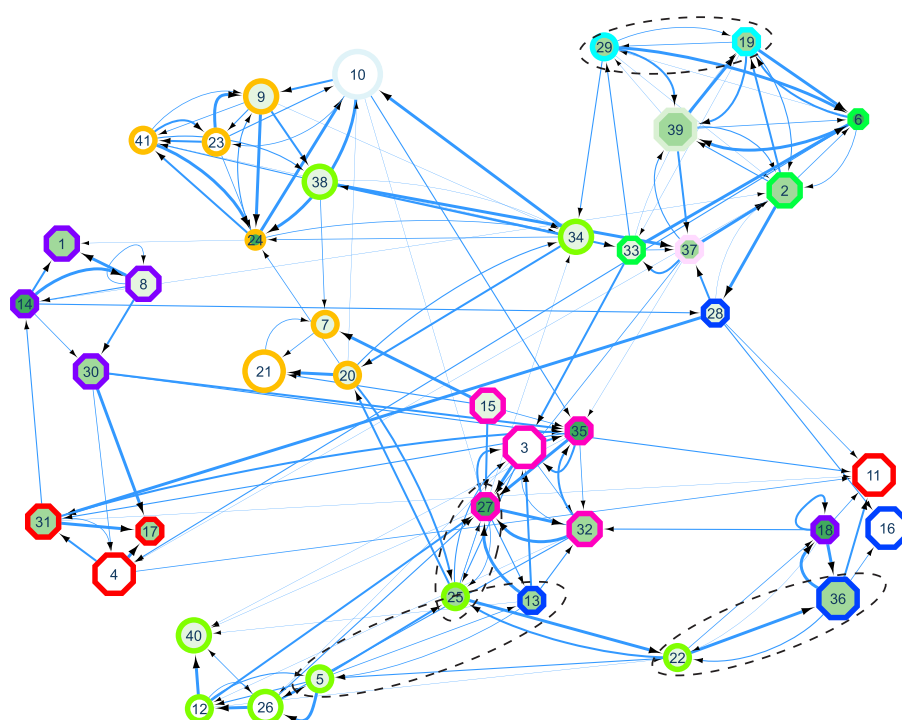


Figure 20 Network of peer interaction among the 41 study abroad sojourners during the final month.

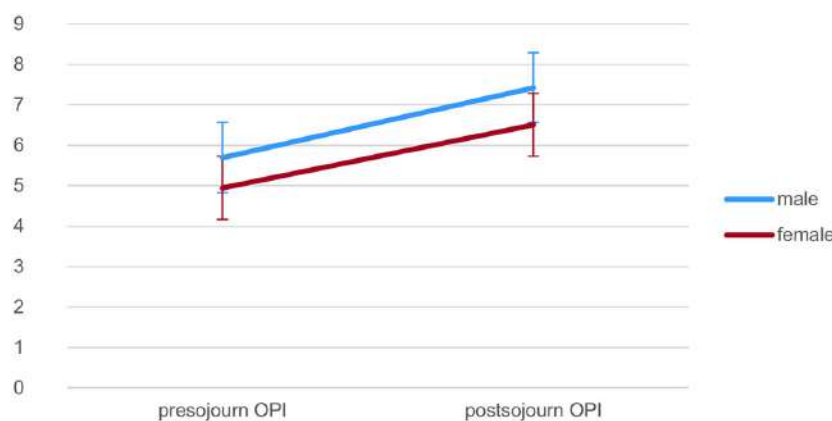


Figure 21 Female versus male students' objective progress as measured by Oral Proficiency Interview (OPI) scores.

the development of written production (Freed et al., 2003; but cf. Pérez-Vidal & Juan-Garau, 2009), where improvement is slower and does not seem to take place until students have spent more than one semester abroad (Sasaki, 2004, 2007, 2009; Serrano et al., 2012).

The middle month of the sojourn revealed a positive relationship between the clustering coefficient, indicating a well-integrated group, and self-perceived progress in culture/pragmatics as well as listening. This degree of group integration may lead to exchange of cultural experiences in the safe space of a familiar community where the learners can feel at home (Gautier & Chevrot, 2015; Ingram, 2005; Schmidt-Rinehart & Knight, 2004), thus playing a role similar to that of pedagogical instruction, which has been shown to be more helpful for pragmatic skills development than exposure to positive evidence alone (Hosseinpur & Nevisi, 2018). A plausible explanation may be that pragmatic knowledge is relatively little affected by the language in which it is conveyed, or alternatively that the sheltered zone of interaction with peers from the sending university prepared the ground for readier interaction with less familiar speakers of Arabic, subsequently helping to develop sociopragmatic competence through social interaction. Exposure to peers' nonnative speech may in turn act as scaffolding towards gradually understanding more idiomatic and dialectal production.

Progress was negatively influenced by the proportion of interaction with alma mater classmates (primarily in the 1st month of the sojourn), as well as by perception of group integration. Although the learners interacted regularly with peers in Arabic, they often used English as well. The anticorrelation (inverse relationship) between progress and percentage of time spent interacting with fellow U.S. classmates replicates findings from other SA research indicating the detrimental influence on language progress of socializing with compatriots and other students sharing the same L1 (Ayano, 2006; Campbell, 2011), even where the participants claimed that a sizable part if not most of the interactions were taking place in the TL (Magnan & Back, 2007; Paradowski et al., 2022). The fact that the 1st month of the sojourn was most influential here underscores the formative impact of the initial period of the stay abroad, and how it may later be difficult to alter the network once it has solidified. These effects are also reflected in the relative stability of the networks. High perception of the integration of one's group may in turn signal lower openness to outsiders. Educating SA participants on the psychological, social, and linguistic effects of interacting with their peers, as suggested by a number of scholars (Coleman & Chafer, 2010; Kelly, 2010; Kinginger, 2008; K. Mitchell, 2012; Ogden, 2007; Sandel, 2014; Wilkinson, 1998), would be beneficial and might facilitate better self-regulation (Bown et al., 2015).

Linguistic Variables and L2 Progress

Prior research has demonstrated that a certain level of proficiency must have been reached for students to benefit from SA (DeKeyser, 2007, 2010; Vande Berg et al., 2009), yet if the level is overly high, measurable gains may be difficult to observe due to a possible ceiling effect (Lapkin et al., 1995) caused by an increasing difficulty for learners in moving up the proficiency levels as they progress up the scale. Our study favors the latter scenario: The students who made the greatest objectively measured progress had lower proficiency at the outset of the sojourn. This mirrors findings from other researchers: Baker-Smemoe et al. (2014), whose participants were the more likely to end up in the “gainer” group the lower they scored on preprogram proficiency; Vande Berg et al. (2009), who relied on simulated OPI scores (similarly to our study) to assess over 800 SA learners; Heinzmann et al. (2024), whose study of 121 students in English-medium instruction contexts (both inner-circle and lingua franca ones—ELFSA) utilized a computerized version of OPI (OPIc); Llanes and Muñoz (2009), who studied 24 Catalan- and Spanish-speaking sojourners in English-speaking countries; and Dewaele and Dewaele (2021), who investigated 33 British and Irish students who spent their SA in Franco-phone countries. Using a numeric score to represent the ACTFL proficiency scale can be problematic because as a learner moves higher up the scale, the amount of language and work required to progress to the next level increases. In actuality, it may be that those who score higher at preprogram but only advance one point or not at all in fact develop *more* linguistically (in terms of total lexis, grammar, pragmatic abilities, etc.) than those who start much lower and move up two or three points. Despite it being common practice, especially in SA research examining predictors of proficiency development (e.g., Davidson, 2010; Segalowitz & Freed, 2004), one should ponder whether the ACTFL proficiency scale should continue to be used as if it were an interval scale (Jelińska & Paradowski, 2021). Finer-grained measures of proficiency gains might be more reliable (see e.g. Zalbidea et al., 2021, p. 261), but potentially also more time-consuming.

Another significant predictor of progress was the students' degree of multilingualism. This finding mirrors one from a cohort of Erasmus+ exchange students in Germany, whose competence in other languages was the strongest correlate of progress ($r = .49$; Paradowski et al., 2022). The literature usually attributes this advantage to influences such as extensive experience with learning form–function mappings (Kaushanskaya, 2012), heightened metalinguistic awareness (Modirghamene, 2006), larger linguistic and intercultural repertoires (Cenoz, 2011), a richer body of knowledge

for beneficial crosslinguistic influence (positive transfer), better established language learning strategies that remove the need to “reinvent the wheel” and lose time on ones that do not work so well for the individual concerned, more realistic expectations of finding both similarities and differences (Paradowski, 2017, p. 236), and/or enhanced executive/cognitive control abilities (Bartolotti & Marian, 2012; Hirosh & Degani, 2018; T. Wang & Saffran, 2014).

While TL use outside of class is widely accepted as a predictor of L2 development, including in SA contexts (Di Silvio et al., 2016; Hernández, 2010; McManus, 2019), in our study it moderately correlated with objective progress among those participants who filled out the survey at all three collection points, but no longer did so when the complete cohort were taken into account, and even in the former group it failed to influence the results once other predictors were considered in the multiple regression and was thus removed from the final model. This emphasizes the importance of realizing “that study abroad is far from a monolithic construct” (Taguchi & Collentine, 2018, p. 564) and of considering a wide array of potential explanatory variables in SA research. Of interest is the fact that this correlation was dominated by the last month of the sojourn. As we have seen, that is the time when L2 use generally drops. This may suggest that students who manage to not give in to this trend, instead persevering with consistent maintenance of practice in the TL, can make more visible strides. Additional support for this claim can be gleaned from the positive influence of the time spent learning Arabic outside of class during the final month of the sojourn.

Dynamic Changes

One dynamic observed in this study is a decrease in participants’ out-of-class L2 use as well as in time spent on language study in the final month of the sojourn. The former mirrors the findings both of García-Amaya (2017), whose analysis of 43 learners enrolled in a 6-week SA program in Salamanca, Spain, showed considerable decrease in rates of L2 use during the last 2 weeks across speaking, listening, writing, and reading, and of Mocanu and Llurda (2020), whose study of 155 international students in Catalonia, Finland, and Romania revealed a drop in motivation to learn the local languages at the end of the sojourners’ stay in all three contexts. The SA literature regularly describes how, as students approach the end of their international experience, they often lose academic focus and try to take advantage of the little remaining time by experiencing the culture, visiting tourist sites, traveling, and so forth (Allen, 2010; Kinginger, 2009). It is also a natural time for students to take time to pack, send packages home, and so on. Numerous activities can serve as distractions from

interacting in the L2 in the last few weeks of the SA experience. The drop in the postsojourn gains could also be due to end-of-program fatigue and staying in “survival mode” just to cross the finish line.

The drop in the percentage of interaction with alma mater classmates during that last month could in turn mean that the students are trying to spend more of their remaining time with locally made friends whom, unlike their peers from their home university, they will no longer be seeing after leaving Jordan. This may be one characteristic qualitatively distinguishing the program at hand from others where sojourners hail from different countries and institutions, and thus are more likely to be missing one another after returning to their respective homelands, which again reiterates the calls against treating SA as a monolithic context.

Past research has suggested that students who make significant gains in the L2 typically start out with relatively large social networks with—necessarily—weak ties, subsequently decreasing the group size while developing stronger and deeper relationships and opportunities for more in-depth and sustained exchanges, this quality aspect overriding the impact of the bare amount of time spent speaking the L2 (Baker-Smemoe et al., 2014; Paradowski et al., 2022). In our cohort, this is noticeable in the decrease in most of the centrality metrics—in-degree (the number of alters speaking to the students), out-degree (the number of alters the students talked to), overall (undirected) degree (number of frequent contacts), closeness, clustering coefficient, and voterank (the measure of optimal influence)—coupled with the steady month-after-month growth in perception of the student group’s integration. The opposite trend of the increase in PageRank—or socializing with likewise “popular” students (Figure 17)—is suggestive of the “rich club phenomenon” (Zhou & Mondragón, 2004) or assortative mixing, where similar (e.g., well-connected or high-achieving) nodes tend to selectively connect to one another.

The drop in self-perceived improvement in reading and vocabulary after the 1st month may suggest that these areas of language see a qualitative jump after arrival in a country that uses a different script. In the USA our participants were likely to have had exposure to Arabic only in class and occasionally in online materials, whereas on arrival in Jordan they were suddenly surrounded with signs and words at every step. After a while, such immersive sink-or-swim effects of the novel linguistic landscape may have normalized.

Subjective progress in listening and grammar in turn saw an improvement in the 3rd month. While these findings merit deeper investigation in future studies, the latter may be the result of a delayed effect of input on interlanguage restructuring. Whereas new lexis may be picked up and put to use immediately

after exposure, it takes decidedly longer to notice, figure out, internalize, and automatize new rules of morphology and syntax.

Combined, the differential trajectories in self-reported progress across skills align with the findings of prior studies (e.g., Serrano et al., 2012), highlighting that (in the SA context) different linguistic subsystems develop at different paces.

Homophily and Gender Differences

Overall, there were minimal observable changes in the makeup of the social networks of learners over the three waves of data collection; although isolated individuals would gravitate from one subgroup to another, on the whole little changed in the composition of the groups. This finding is in line with the general stability and lack of change in social network characteristics over the SA sojourn reported by McManus (2019) for British university students spending an academic year in France, and Strawbridge (2023a) for U.S. students during a one-semester stay in Spain. One exception, however, is the pattern of males and females forming separate groups, a division that strengthened over time. The tendency to associate with those who are like oneself (homophily) is commonplace in education, starting in elementary and secondary school (Shrum et al., 1988) and continuing into higher education (Weber et al., 2020), and in society in general (McPherson et al., 2001). Weber et al. (2020) note that gender homophily is more likely when network members are relatively homogeneous in their academic abilities, a description that to some degree matches the Arabic learners in this group: they were largely high-achieving students, were required to receive B or better grades in preparation for the SA program, and all came from the same home university, but they did differ considerably in their Arabic proficiency coming into the program.

An important note here is that, in alignment with Jordanian society's tendency to segregate men and women, the women (save for two married female students) shared housing with each other, as did the single men. The societal expectation could be responsible for the unease over intermixing of genders in public. For instance, we see that the female student (Node 29) who stayed closest to an all-male group (of whom all but Node 28 were married and accompanied by a spouse in Jordan) did so because she was married to and lived with her husband (Node 19). Figures 18–20 show how the only other female student who managed to gain access to and be part of that group was Node 29's friend (Node 34), and even then she did so only in the 1st month, while later maintaining a connection only through Node 29. This replicates Dewey, Ring et al.'s (2013) finding that U.S. students of Arabic in Egypt and Jordan

typically spoke with locals of the same sex, especially when considering closer friends.

At any time of their sojourn, the participants in this program were spending on average between 56.6% and 59% of their total interaction with their U.S. peers. This tendency of (especially American) SA sojourners to retreat into the comfort of conational or international “bubbles” of fellow program participants has long been commented on in the SA literature (e.g., Beech, 2018; Bracke & Aguerre, 2015; Brown, 2008, 2009; Church, 1982; Coleman & Chafer, 2010; Furnham & Alibhai, 1985; Hendrickson, 2018; Kelly, 2010; Kinginger, 2008; Mikal et al., 2015; K. Mitchell, 2012; Ogden, 2007; Sandel, 2014; Paradowski et al., 2022; Van Mol & Michielsens, 2014; Wright & Scharfner, 2013; Zhou & Rose, 2023). For instance, Strawbridge (2023b), in a study of 30 U.S. learners enrolled in semester-long university SA programs in Spain, found that 89% of all the participant networks he investigated were centered around a small number of dense cliques made up of other non-native-speaker program peers.

In the case of our study, women were more likely to display a higher proportion of their communication with their alma mater classmates than did their male colleagues, especially during the 1st month of their sojourn. At least for some, interacting with peers probably constituted the “formation of home culture ‘islands’” and may have been “the most efficient way for the students to keep from drowning in the [Jordanian] ‘Ocean’” while they processed “the barrage of cultural differences and linguistic challenges faced on a daily basis” (Wilkinson, 1998, p. 32), resorting to the decompression pattern to help deal with homesickness and improve well-being. When drowning in the unfamiliar or struggling with mental health issues (as some did in our study according to the reports of the program coordinator), seeking contact with compatriots—whether on site or back at home—may have restorative value, and need not necessarily affect developing ties with host community members (Savicki, 2010). Needless to say, the women in this study may have looked for support among their largely female peers. This is not surprising given that research on coping with mental health challenges indicates females are more likely than males to seek support in facing challenges (Liddon et al., 2018).

Also partly responsible for women’s higher percentage of interaction with fellow U.S. classmates may have been their propensity to share their accommodation with more peers, but there is the cultural dimension to be considered as well. Prior research involving SA participants in the Middle East (Bown et al., 2015; Trentman, 2015), as well as other areas such as Russia (Pellegrino Aveni, 2005), Costa Rica (Twombly, 1995), and France (Kinging, 2008), has shown

that women can face challenges in creating new social relationships with locals in the public sphere, but have the advantage over men when it comes to access to women and to family life. For previous female students, sexual harassment had often been a significant challenge, resulting in the need to seek “refuge in a study abroad peer group” (Trentman, 2015, p. 267). Bown et al. (2015) characterized this behavior as a form of self-regulation, whereby the women “actively shape the learning environment, creating experiences that produce the conditions deemed necessary for meeting their goals” (p. 216).

As ascertained from discussion with the course coordinator, who drew on their daily observation of and conversations with the students, the female students in this particular cohort appeared to have had far fewer negative experiences of this nature, for instance feeling relatively comfortable using Uber taxis. However, going out in the evening was still more of a challenge for them than for their male classmates, and some did not feel they had the social capital to regularly ask men to accompany them. Many men and women did not form close friendships with locals that would have provided a natural opportunity to get more speaking time with them. The result was felt more acutely by the women, as men could easily engage in “cold contacting,” the most common means of finding people to talk with and potentially making friends reported by Dewey, Ring, et al. (2013). In short, although sexual harassment did not play the role it had previously played for students in a variety of Arab countries, traditional Arab culture was nevertheless an impediment to finding speaking opportunities for many of the female students. This also underscores the methodological importance of considering the local context in accounting for observed phenomena.

The inverse relationship between progress and presojourn proficiency in Arabic remained strong both in the entire cohort and after the data were split by gender. Still, on the whole the female participants ended up with lower final OPI scores than men (despite greater time speaking Arabic with classmates), though their self-ratings suggest that they felt they were on average more proficient than their peers. This comes as little surprise, given that the female students on average arrived in Jordan with weaker speaking ability, and would have perceived the change from quite halting communication to more automatic facility as considerable, compared to the less palpable but objectively still comparable progress made by the more advanced male students.

Limitations and Future Directions

This study comes with obvious limitations, such as reliance on self-reports and on students’ ability to accurately quantify the magnitude of their social in-

teractions, as well as a lack of personality measures. The former is a necessity in virtually all SA research that recreates students' social networks (see, e.g., Dewey, 2017); the latter is the result of the natural trade-off between comprehensiveness and feasibility. Also, not all the participants in the current study filled out all three of the monthly questionnaires, which lowers the power of the explanatory model; its instability shows when bias-corrected and accelerated (BCa) bootstrapping or alternative covariance matrix estimators are used, as these occasionally flip the significance of some of the predictors in the regression (notably in-degree and degree of multilingualism). A power analysis for a model with five predictors suggests a sample size of 89 participants to obtain a medium effect size of $f^2 = .15$ (G*Power; Faul et al., 2009). Such group sizes are unrealistic for whole-network analyses of language learner cohorts, but to address the power issue, future researchers could try incorporating data from successive iterations of the same SA program, alongside attempting to ensure consistently higher completion rates across time points.

Additionally, in the case of smaller student groups, to address possible recall bias in students' self-reports of language use and social interactions, the whole-network approach could be complemented with, for instance, the experience sampling method (Csíkszentmihályi, Larson, & Prescott, 1977), which involves longitudinally intensive soliciting of short here-and-now behavioral reports, such as that exemplified in applied linguistics by studies using the Lang-Track-App (Arndt et al., 2023a, 2023b). Additional insights—for example, into the dynamics—could be gleaned from mixed-methods approaches involving semistructured participant interviews or language logs, especially focusing on the initial and final weeks of the sojourn.

Methodologically, the present study extends recent work (e.g., Gautier, 2019; Hasegawa, 2019; McManus, 2019; Strawbridge, 2023a) that has begun to evidence longitudinal changes in sojourners' social network development during study abroad. The application of a dynamic computational social network analysis reconstructing whole learner networks to longitudinally investigate study-abroad SLA reveals the potential of the method to answer both long-standing and novel research questions regarding the relationship between students' peer interactions and their linguistic progress. The visible changes in the patterns and strength of social interactions despite the relative stability in terms of participants' positions in the graph underscore the importance of adopting a longitudinal lens with repeated measurement points, as emphasized by Gautier (2019), Gautier and Chevrot (2015), and Strawbridge (2023a). Future quantitative analyses of this nature may complement the picture with investigations of more variables potentially influencing the patterns of social be-

havior and trajectories of sojourners' social networks in the first place, as well as algorithms such as cluster/community detection (Paradowski et al., 2022).

Conclusions

Students do not always form sufficient social bonds to support their L2 learning, whether with classmates or locals. Given the clear importance of such ties for L2 use, students could benefit from the assistance of SA programs in encouraging and creating such relationships, through the organization of integration events and the provision of other structured opportunities (Baker-Smemoe et al., 2014; Chamot et al., 2021; McManus, 2019; Regan et al., 2009); they might also benefit from properly informative predeparture programs (Moreno Bruna & Goethals, 2020). This type of scaffolding could help learners to gain confidence both in their L2 abilities and in their capacity to build social connections in the TL. At the same time, awareness of the relationships between social network patterns and L2 gains could have direct utility for program coordinators in action research, helping them to identify students at risk of lower gains after the first round of social network data collection, provided that the students consented to such use of the information (see also Strawbridge, 2023b, p. 1132).

In considering the nonlinear interaction of a variety of linguistic and extralinguistic variables in L2 development, this study can also be positioned in the perspective of dynamic systems theory (DST; e.g., Lowie et al., 2009). In addition, it contributes to the still small body of literature on the acquisition of less commonly taught languages (LCTLs; especially languages other than English—LOTE). Lastly, the findings underscore the importance of recognizing the ecological context of (study-abroad) language learning, here manifested, for instance, in the cultural expectations concerning living arrangements and in logistic constraints.

Final revised version accepted 8 July 2024

Notes

- 1 We use both abbreviations in order to avoid repetition of the same labels, in line with existing literature much of which has focused on contexts of actual *second* language acquisition, in contrast to studies where the language being acquired was not necessarily the second but sometimes the third, fourth, or further language of the participants (as was the case for many of the students in the study at hand). We adopt this usage also in order to steer away from potentially misleading terms such as “L2 speakers” where both native and nonnative speakers of the TL are meant.

- 2 Although it could have yielded interesting insights, we did not separate out a third group of students living with their families given too low a number thereof ($n = 8$), not sufficient to yield adequately robust relationships in the data.
- 3 Such studies may offer insights given that, as remarked by one of the reviewers, such opportunities for contact with TL speakers are often hard to come by.
- 4 The self-ratings of participants' proficiency in other languages utilized the CEFR scale (Council of Europe, 2020). As most of the students were not familiar with these levels, equivalents were given in ACTFL Oral Proficiency Interview (OPI) ratings during the explanation of the study and survey (these ratings are set by the American Council on the Teaching of Foreign Languages). A few students in the program had received a language certificate after taking an OPI in Arabic and therefore already had some familiarity with what speakers should be able to do at each level. The ACTFL OPI scales were then additionally explained in terms of abilities.
- 5 Network Diffusion: https://github.com/anty-filidor/network_diffusion
- 6 NetworkX: <https://networkx.org>
- 7 Matplotlib: <https://matplotlib.org>
- 8 Cytoscape: <https://cytoscape.org>
- 9 The scoring is based on a holistic rating encompassing the learner's performance on content, tasks, functions, and accuracy; see <http://www.language-testing.com> for details on the interview and scores.
- 10 In the reporting of correlations, the values report sequentially Pearson's r (no letter) or Spearman's rho (r_s , where at least one of the variables was skewed) and bias-corrected 95% confidence interval.

References

- Allen, H. (2010). Interactive contact as linguistic affordance during short-term study abroad: Myth or reality? *Frontiers: The Interdisciplinary Journal of Study Abroad*, 19(1), 1–26. <https://doi.org/10.36366/frontiers.v19i1.271>
- Arndt, H. L., Granfeldt, J., & Gullberg, M. (2023a). Reviewing the potential of the Experience Sampling Method (ESM) for capturing second language exposure and use. *Second Language Research*, 39(1), 39–58. <https://doi.org/10.1177/02676583211020055>
- Arndt, H. L., Granfeldt, J., & Gullberg, M. (2023b). The Lang-Track-App: Open-source tools for implementing the Experience Sampling Method in second language acquisition research. *Language Learning*, 73(3), 869–903. <https://doi.org/10.1111/lang.12555>
- Arnett, C. (2013). Syntactic gains in short-term study abroad. *Foreign Language Annals*, 46(4), 705–712. <https://doi.org/10.1111/flan.12052>

- Atkinson, D. (2002). Toward a sociocognitive approach to second language acquisition. *The Modern Language Journal*, 86(4), 525–545. <https://doi.org/10.1111/1540-4781.00159>
- Atsuzawa-Windley, S., & Noguchi, S. (1995). Effects of in-country experience on the acquisition of oral communication skills in Japanese. *Australian Review of Applied Linguistics*, 12(1), 83–98. <https://doi.org/10.1075/aralss.12.06ats>
- Ayano, M. (2006). Japanese students in Britain. In M. Byram & A. Feng (Eds.), *Living and studying abroad: Research and practice* (pp. 11–37). Multilingual Matters. <https://doi.org/10.21832/9781853599125-004>
- Baker-Smemoe, W., Dewey, D. P., Bown, J., & Martinsen, R. A. (2014). Variables affecting L2 gains during study abroad. *Foreign Language Annals*, 47(3), 464–486. <https://doi.org/10.1111/flan.12093>
- Barkaoui, K. (2014). Quantitative approaches for analysing longitudinal data in second language research. *Annual Review of Applied Linguistics*, 34, 65–101. <https://doi.org/10.1017/S0267190514000105>
- Barron, A. (2002). *Acquisition in interlanguage pragmatics: Learning how to do things with words in a study abroad context*. John Benjamins. <https://doi.org/10.1075/pbns.108>
- Bartolotti, J., & Marian, V. (2012). Language learning and control in monolinguals and bilinguals. *Cognitive Science*, 36(6), 1129–1147. <https://doi.org/10.1111/j.1551-6709.2012.01243.x>
- Bataller, R. (2010). Making a request for a service in Spanish: Pragmatic development in the study abroad setting. *Foreign Language Annals*, 43(1), 160–175. <https://doi.org/10.1111/j.1944-9720.2010.01065.x>
- Baten, K. (2020). The role of social networks and intense friendships in study abroad students' L2 use and speaking development: Erasmus learners' perspectives. In A. M. Devlin (Ed.), *Study abroad and the Erasmus+ programme in Europe: Perspectives on language and intercultural learning* (pp. 15–44). John Benjamins. <https://doi.org/10.1075/sar.18009.bat>
- Beech, S. E. (2018). Negotiating the complex geographies of friendships overseas: Becoming, being and sharing in student mobility. *Geoforum*, 92, 18–25. <https://doi.org/10.1016/j.geoforum.2018.03.019>
- Bird, M. T., & Belnap, R. K. (2018). Lessons from 25 years of experimenting with an Arabic study abroad program. In J. L. Plews & K. Misfeldt (Eds.), *Second language study abroad: Programming, pedagogy, and participant engagement* (pp. 49–82). Palgrave Macmillan. https://doi.org/10.1007/978-3-319-77134-2_3
- Borgatti, S. P., Everett, M. G., Johnson, J. C., & Agneessens, F. (2024). *Analyzing social networks* (3rd ed.). SAGE.
- Borràs, J., & Llanes, À. (2019). Re-examining the impact of study abroad on L2 development: A critical overview. *The Language Learning Journal*, 49(5), 527–540. <https://doi.org/10.1080/09571736.2019.1642941>

- Bourdieu, P. (1986). The forms of capital. In J. Richardson (Ed.), *Handbook of theory and research for the sociology of education* (pp. 241–258). Greenwood.
- Bown, J., Dewey, D. P., & Belnap, R. K. (2015). Student interactions during study abroad in Jordan. In R. Mitchell, N. Tracy-Ventura, & K. McManus (Eds.), *Social interaction, identity and language learning during residence abroad* (pp. 199–222). European Second Language Association.
https://www.eurosla.org/monographs/EM04/Bown_etal.pdf
- Bracke, A., & Aguerre, S. (2015). Erasmus students: Joining communities of practice to learn French? In R. Mitchell, N. Tracy-Ventura, & K. McManus (Eds.), *Social interaction, identity and language learning during residence abroad* (pp. 139–168). European Second Language Association.
https://www.eurosla.org/monographs/EM04/Bracke_Aguerre.pdf
- Brecht, R., Davidson, D., & Ginsberg, R. (1995). Predictors of foreign language gain during study abroad. In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 37–66). John Benjamins. <https://doi.org/10.1075/sibil.9.05bre>
- Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7), 107–117.
[https://doi.org/10.1016/S0169-7552\(98\)00110-X](https://doi.org/10.1016/S0169-7552(98)00110-X)
- Brown, L. (2008). Language and anxiety: An ethnographic study of international postgraduate students. *Evaluation & Research in Education*, 21(2), 75–95.
<https://doi.org/10.1080/09500790802152167>
- Brown, L. (2009). An ethnographic study of the friendship patterns of international students in England: An attempt to recreate home through conational interaction. *International Journal of Educational Research*, 48(3), 184–193.
<https://doi.org/10.1016/j.ijer.2009.07.003>
- Bródka, P., Saganowski, S., & Kazienko, P. (2013). GED: The method for group evolution discovery in social networks. *Social Network Analysis and Mining*, 3, 1–14. <https://doi.org/10.1007/s13278-012-0058-8>
- Campbell, R. (2011). The impact of study abroad on Japanese language learners' social networks. *New Voices*, 5, 25–63. <https://doi.org/10.21159/nv.05.02>
- Carhill-Poza, A., & Kurata, N. (2020). *Social networks in language learning and language teaching*. Bloomsbury.
- Cartier van Dissel, M., Góra, P., Iskrzyński, M., Kramar Fijavž, M., Manea, D., Mauroy, A., Nakić, I., Nicaise, S., Paradowski, M. B., Rotundo, G., Puchalska, A., & Sikolya, E. (in press). Journey through the world of dynamical systems on networks. In M. Čolić, J. Giesselmann, J. Glück, M. Kramar Fijavž, D. Mugnolo, & A. Mauroy (Eds.), *Mathematical models for interacting dynamics on networks*. Springer.
- Cenoz, J. (2011). The influence of bilingualism on third language acquisition: Focus on multilingualism. *Language Teaching*, 46(1), 71–86.
<https://doi.org/10.1017/S0261444811000218>

- Chamot, M., Racine, I., Regan, V., & Detey, S. (2021). Une ou des immersion(s) ? Regard sur l'acquisition de la compétence sociolinguistique par des apprenants anglophones irlandais de FLE. In E. Pustka (Ed.), *La prononciation du français langue étrangère : Perspectives linguistiques et didactiques* (pp. 133–161). Narr.
- Chaudron, C. (1988). *Second language classrooms: Research on teaching and learning*. Cambridge University Press. <https://doi.org/10.1017/CBO9781139524469>
- Church, A. T. (1982). Sojourner adjustment. *Psychological Bulletin*, 91(3), 540–572. <https://doi.org/10.1037/0033-2909.91.3.540>
- Coleman, J. A. (2015). Social circles during residence abroad: What students do, and who with. In R. Mitchell, N. Tracy-Ventura, & K. McManus (Eds.), *Social interaction, identity and language learning during residence abroad* (pp. 33–52). European Second Language Association. <https://www.eurosla.org/monographs/EM04/Coleman.pdf>
- Coleman, J. A., & Chafer, T. (2010). Study abroad and the Internet: Physical and virtual context in an era of expanding telecommunications. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 19(1), 151–168. <https://doi.org/10.36366/frontiers.v19i1.278>
- Collentine, J. (2004). The effects of learning contexts on morphosyntactic and lexical development. *Studies in Second Language Acquisition*, 26(2), 227–248. <https://doi.org/10.1017/S0272263104262040>
- Collentine, J., & Freed, B. F. (2004). Learning context and its effects on second language acquisition: Introduction. *Studies in Second Language Acquisition*, 26(2), 153–172. <https://doi.org/10.1017/S0272263104262015>
- Council of Europe. (2020). *Common European Framework of Reference for Languages: Learning, teaching, assessment – Companion volume*. Council of Europe. <https://rm.coe.int/common-european-framework-of-reference-for-languages-learning-teaching/16809ea0d4>
- Cribari-Neto, F. (2004). Asymptotic inference under heteroskedasticity of unknown form. *Computational Statistics & Data Analysis*, 45(2), 215–233. [https://doi.org/10.1016/S0167-9473\(02\)00366-3](https://doi.org/10.1016/S0167-9473(02)00366-3)
- Csikszentmihályi, M., Larson, R., & Prescott, S. (1977). The ecology of adolescent activity and experience. *Journal of Youth and Adolescence*, 6, 281–294. <https://doi.org/10.1007/BF02138940>
- Czuba, M., & Bródka, P. (2022). Simulating spreading of multiple interacting processes in complex networks. In J. Z. Huang, Y. Pan, B. Hammer, M. K. Khan, X. Xie, L. Cui, & Y. He (Eds.), *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA), Shenzhen, China, 2022* (pp. 1–10). IEEE Computer Society. <https://doi.org/10.1109/DSAA54385.2022.10032425>
- Davidson, D. (2010). Study abroad: When, how long, and with what results? New data from the Russian front. *Foreign Language Annals*, 43(1), 6–26. <https://doi.org/10.1111/j.1944-9720.2010.01057.x>

- de Bot, K. (1996). The psycholinguistics of the output hypothesis. *Language Learning*, 46(3), 529–555. <https://doi.org/10.1111/j.1467-1770.1996.tb01246.x>
- DeKeyser, R. M. (1991). Foreign language development during a semester abroad. In B. F. Freed (Ed.), *Foreign language acquisition research and the classroom* (pp. 104–119). D. C. Heath.
- DeKeyser, R. M. (2007). Study abroad as foreign language practice. In R. M. DeKeyser (Ed.), *Practice in a second language: Perspectives from applied linguistics and cognitive psychology* (pp. 208–226). Cambridge University Press.
- DeKeyser, R. (2010). Monitoring processes in Spanish as a second language during a study abroad program. *Foreign Language Annals*, 43(1), 80–92. <https://doi.org/10.1111/j.1944-9720.2010.01061.x>
- DeKeyser, R. (2014). Research on language development during study abroad: Methodological considerations and future perspectives. In C. Pérez-Vidal (Ed.), *Language acquisition in study abroad and formal acquisition contexts* (pp. 313–326). John Benjamins. <https://doi.org/10.1075/aals.13.16ch13>
- Devlin, A. M. (2019). The interaction between duration of study abroad, diversity of loci of learning and sociopragmatic variation patterns: A comparative study. *Journal of Pragmatics*, 146, 121–136. <https://doi.org/10.1016/j.pragma.2018.08.007>
- Dewaele, L., & Dewaele, J.-M. (2021). Actual and self-perceived linguistic proficiency gains in French during study abroad. *Languages*, 6(1), Article 6. <https://doi.org/10.3390/languages6010006>
- Dewey, D. P. (2004). A comparison of reading development by learners of Japanese in intensive and domestic immersion and study abroad contexts. *Studies in Second Language Acquisition*, 26(2), 303–327. <https://doi.org/10.1017/S0272263104262076>
- Dewey, D. P. (2017). Measuring social interaction during study abroad: Quantitative methods and challenges. *System*, 71, 49–59. <https://doi.org/10.1016/j.system.2017.09.026>
- Dewey, D. P., Belnap, R. K., & Hillstrom, R. (2013). Social network development, language use, and language acquisition during study abroad: Arabic language learners' perspectives. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 22(1), 84–110. <https://doi.org/10.36366/frontiers.v22i1.320>
- Dewey, D. P., Bown, J., Baker, W., Martinsen, R. A., Gold, C., & Eggett, D. (2014). Language use in six study abroad programs: An exploratory analysis of possible predictors. *Language Learning*, 64(1), 36–71. <https://doi.org/10.1111/lang.12031>
- Dewey, D. P., Bown, J., & Eggett, D. (2012). Japanese language proficiency, social networking, and language use during study abroad: Learners' perspectives. *The Canadian Modern Language Review*, 68(2), 111–137. <https://doi.org/10.3138/cmlr.68.2.111>
- Dewey, D., Ring, S., Gardner, D., & Belnap, K. (2013). Social network formation and development during study abroad in the Middle East. *System*, 41(2), 269–282. <https://doi.org/10.1016/j.system.2013.02.004>

- Diao, W., & Maa, J. (2019). Language socialization and L2 pragmatics. In N. Taguchi (Ed.), *The Routledge handbook of second language acquisition and pragmatics* (pp. 128–144). Routledge. <https://doi.org/10.4324/9781351164085-9>
- Díaz-Campos, M. (2004). Context of learning in the acquisition of Spanish second language phonology. *Studies in Second Language Acquisition*, 26(2), 249–273. <https://doi.org/10.1017/S0272263104262052>
- Di Silvio, F., Diao, W., & Donovan, A. (2016). The development of L2 fluency during study abroad: A cross-language study. *The Modern Language Journal*, 100(3), 610–624. <https://doi.org/10.1111/modl.12343>
- Dogrusoz, U., Giral, E., Cetintas, A., Civril, A., & Demir, E. (2009). A layout algorithm for undirected compound graphs. *Information Sciences*, 179(7), 980–994. <https://doi.org/10.1016/j.ins.2008.11.017>
- Duff, P. A. (2007). Second language socialization as sociocultural theory: Insights and issues. *Language Teaching*, 40(4), 309–319. <https://doi.org/10.1017/S0261444807004508>
- Duff, P. A. (2019). Social dimensions and processes in second language acquisition: Multilingual socialization in transnational contexts. *The Modern Language Journal*, 103(S1), 6–22. <https://doi.org/10.1111/modl.12534>
- Ellis, R. (2021). A short history of SLA: Where have we come from and where are we going? *Language Teaching*, 54(2), 190–205. <https://doi.org/10.1017/S0261444820000038>
- Faul, F., Erdfelder, E., Buchner, A., & Lang, A.-G. (2009). Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. *Behavior Research Methods*, 41, 1149–1160. <https://doi.org/10.3758/BRM.41.4.1149>
- Foster, G. C., Lane, D., Scott, D., Hebl, M., Guerra, R., Osherson, D., & Zimmer, H. (2018). *An introduction to psychological statistics*. Open Educational Resources Collection. 4. <https://irl.umsl.edu/oer/4>
- Freed, B. F. (1995). What makes us think that students who study abroad become fluent? In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 123–148). John Benjamins. <https://doi.org/10.1075/sibil.9.09fre>
- Freed, B. F., Dewey, D. P., Segalowitz, N., & Halter, R. (2004). The language contact profile. *Studies in Second Language Acquisition*, 26(2), 349–356. <https://doi.org/10.1017/S027226310426209X>
- Freed, B. F., So, S., & Lazar, N. A. (2003). Language learning abroad: How do gains in written fluency compare with gains in oral fluency in French as a second language? *ADFL Bulletin*, 34(3), 34–40. <https://doi.org/10.1632/adfl.34.3.34>
- Furnham, A., & Alibhai, N. (1985). The friendship networks of foreign students: A replication and extension of the functional model. *International Journal of Psychology*, 20(3–4), 709–722. <https://doi.org/10.1080/00207598508247565>
- García-Amaya, L. (2017). Detailing L1 and L2 use in study-abroad research: Data from the daily linguistic questionnaire. *System*, 71, 60–72. <https://doi.org/10.1016/j.system.2017.09.023>

- Gass, S. M. (2003). Input and interaction. In C. J. Doughty & M. H. Long (Eds.), *The handbook of second language acquisition* (pp. 224–255). Blackwell.
<https://doi.org/10.1002/9780470756492.ch9>
- Gass, S. M. (2010). Interactionist perspectives on second language acquisition. In R. B. Kaplan (Ed.), *The Oxford handbook of applied linguistics* (2nd ed., pp. 1–18). Oxford University Press.
<https://doi.org/10.1093/oxfordhb/9780195384253.013.0015>
- Gautier, R. (2019). Understanding socialization and integration through social network analysis: American and Chinese students during a stay abroad. In M. Howard (Ed.), *Study abroad, second language acquisition and interculturality* (pp. 207–236). Multilingual Matters. <https://doi.org/10.21832/9781788924153-011>
- Gautier, R., & Chevrot, J. (2015). Acquisition of sociolinguistic variation in a study abroad context: The impact of social network. In R. Mitchell, N. Tracy-Ventura, & K. McManus (Eds.), *Social interaction, identity and language learning during residence abroad* (pp. 169–184). European Second Language Association.
http://www.eurosla.org/monographs/EM04/Gautier_Chevrot.pdf
- Geeslin, K., Fafulus, S., & Kanwit, M. (2013). Acquiring geographically-variable norms of use: The case of the present perfect in Mexico and Spain. In C. Howe, S. E. Blackwell, & M. L. Quesada (Eds.), *Selected proceedings of the 15th Hispanic Linguistics Symposium* (pp. 205–220). Cascadilla Proceedings Project.
- Geeslin, K. L., & Long, A. Y. (2014). *Sociolinguistics and second language acquisition: Learning to use language in context*. Routledge.
<https://doi.org/10.4324/9780203117835>
- Goldoni, F. (2013). Students' immersion experiences in study abroad. *Foreign Language Annals*, 46(3), 359–376. <https://doi.org/10.1111/flan.12047>
- Grey, S., Cox, J. G., Serafini, E. J., & Sanz, C. (2015). The role of individual differences in the study abroad context: Cognitive capacity and language development during short-term intensive language exposure. *The Modern Language Journal*, 99(1), 137–157. <https://doi.org/10.1111/modl.12190>
- Gürüz, K. (2011). *Higher education and international student mobility in the global knowledge economy* (Revised and updated 2nd ed.). State University of New York Press.
- Hasegawa, A. (2019). *The social lives of study abroad: Understanding second language learners' experiences through social network analysis and conversation analysis*. Routledge. <https://doi.org/10.4324/9780429505829>
- Hayes, A. F., & Cai, L. (2007). Using heteroskedasticity-consistent standard error estimators in OLS regression: An introduction and software implementation. *Behavior Research Methods*, 39, 709–722. <https://doi.org/10.3758/BF03192961>
- Heinzmann, S., Hilbe, R., Ehram, K., & Bleichenbacher, L. (2024). SA students' social contacts in different linguistic contexts and their relationship with English use and development. *Language Learning*, 74(S2), 24–57.
<https://doi.org/10.1111/lang.12674>

- Hendrickson, B. (2018). Intercultural connectors: Explaining the influence of extra-curricular activities and tutor programs on international student friendship network development. *International Journal of Intercultural Relations*, 63, 1–16. <https://doi.org/10.1016/j.ijintrel.2017.11.002>
- Hernández, T. A. (2010). The relationship among motivation, interaction, and the development of second language oral proficiency in a study-abroad context. *The Modern Language Journal*, 94(4), 600–617. <https://doi.org/10.1111/j.1540-4781.2010.01053.x>
- Hirosh, Z., & Degani, T. (2018). Direct and indirect effects of multilingualism on novel language learning: An integrative review. *Psychonomic Bulletin & Review*, 25, 892–916. <https://doi.org/10.3758/s13423-017-1315-7>
- Hosseinpur, R. M., & Nevisi, R. B. (2018). L1-based instruction: Does it work for learning pragmatics in the EFL context? *The Journal of Asia TEFL*, 15(1), 233–241. <https://doi.org/10.18823/asiatefl.2018.15.1.18.233>
- Howard, M. (2011). Input perspectives on the role of learning context in Second Language Acquisition: An introduction to the special issue. *IRAL – International Review of Applied Linguistics in Language Teaching*, 49(2), 71–82. <https://doi.org/10.1515/iral.2011.004>
- Huensch, A., & Tracy-Ventura, N. (2017). L2 utterance fluency development before, during, and after residence abroad: A multidimensional investigation. *The Modern Language Journal*, 101(2), 275–293. <https://doi.org/10.1111/modl.12395>
- Ingram, M. (2005). Recasting the foreign language requirement through study abroad: A cultural immersion program in Avignon. *Foreign Language Annals*, 38(2), 211–222. <https://doi.org/10.1111/j.1944-9720.2005.tb02486.x>
- Isabelli-García, C. L. (2006). Study abroad social networks, motivation and attitudes: Implications for second language acquisition. In M. A. DuFon & E. Churchill (Eds.), *Language learners in study abroad contexts* (pp. 231–258). Multilingual Matters. <https://doi.org/10.21832/9781853598531-013>
- Isabelli-García, C. (2010). Acquisition of Spanish gender agreement in two learning contexts: Study abroad and at home. *Foreign Language Annals*, 43(2), 289–303. <https://doi.org/10.1111/j.1944-9720.2010.01079.x>
- Isabelli-García, C., Bown, J., Plews, J. L., & Dewey, D. P. (2018). Language learning and study abroad. *Language Teaching*, 51(4), 439–484. <https://doi.org/10.1017/S026144481800023X>
- Issa, B. I., Faretta-Stutenberg, M., & Bowden, H. W. (2020). Grammatical and lexical development during short-term study abroad: Exploring L2 contact and initial proficiency. *The Modern Language Journal*, 104(4), 860–879. <https://doi.org/10.1111/modl.12677>
- Iwasaki, N. (2010). Style shifts among Japanese learners before and after study abroad in Japan: Becoming active social agents in Japanese. *Applied Linguistics*, 31(1), 45–71. <https://doi.org/10.1093/applin/amn047>

- Jackson, J. (2017). Intervening in the intercultural learning of L2 study abroad students: From research to practice. *Language Teaching*, 51(3), 365–382. <https://doi.org/10.1017/S0261444816000392>
- Jelińska, M., & Paradowski, M. B. (2021, Jul 10–14). *Multilingualism correlates with teachers' coping with emergency remote instruction, but this depends on the measure: Methodological and practical implications* [Paper presentation]. International Symposium on Bilingualism (ISB13) “Bilingualism in Flux,” University of Warsaw, Poland.
- Juan-Garau, M. (2018). Exploring oral L2 fluency development during a three-month stay abroad through a dialogic task. In C. Sanz & A. Morales-Front (Eds.), *The Routledge handbook of study abroad research and practice* (pp. 193–207). Routledge. <https://doi.org/10.4324/9781315639970-13>
- Kaushanskaya, M. (2012). Cognitive mechanisms of word learning in bilingual and monolingual adults: The role of phonological memory. *Bilingualism: Language and Cognition*, 15(3), 470–489. <https://doi.org/10.1017/S1366728911000472>
- Kelly, D. (2010). Student learning in an international setting. *New Directions for Higher Education*, 150, 97–107. <https://doi.org/10.1002/he.393>
- Kennedy Terry, K. (2017). Contact, context, and collocation: The emergence of sociostylistic variation in L2 French learners during study abroad. *Studies in Second Language Acquisition*, 39(3), 553–578. <https://doi.org/10.1017/S0272263116000061>
- Kennedy Terry, K. (2022a). Social networks. In K. Geeslin (Ed.), *The Routledge handbook of sociolinguistics and second language acquisition* (pp. 113–125). Routledge. <https://doi.org/10.4324/9781003017325-11>
- Kennedy Terry, K. (2022b). Sociolinguistic variation in L2 French: What schwa deletion patterns reveal about language acquisition during study abroad. In R. Bayley, D. R. Preston, & X. Li (Eds.), *Variation in second and heritage languages: Crosslinguistic perspectives* (pp. 279–310). John Benjamins. <https://doi.org/10.1075/silv.28.11ter>
- Kim, H.-Y. (2013). Statistical notes for clinical researchers: Assessing normal distribution using skewness and kurtosis. *Restorative Dentistry and Endodontics*, 38(1), 52–54. <https://doi.org/10.5395/rde.2013.38.1.52>
- Kinginger, C. (2008). Language learning in study abroad: Case studies of Americans in France. *The Modern Language Journal*, 92(1), 1–124. <https://doi.org/10.1111/j.1540-4781.2008.00821.x>
- Kinginger, C. (2009). *Language learning and study abroad: A critical reading of research*. Palgrave Macmillan. <https://doi.org/10.1057/9780230240766>
- Kinginger, C. (2011). Enhancing language learning in study abroad. *Annual Review of Applied Linguistics*, 31, 58–73. <https://doi.org/10.1017/S0267190511000031>
- Kinginger, C. (2013). Identity and language learning in study abroad. *Foreign Language Annals*, 46(3), 339–358. <https://doi.org/10.1111/flan.12037>

- Kinginger, C., & Belz, J. A. (2005). Sociocultural perspectives on pragmatic development in foreign language learning: Case studies from telecollaboration and study abroad. *Intercultural Pragmatics*, 2(4), 369–421.
<https://doi.org/10.1515/iprg.2005.2.4.369>
- Kinginger, C., & Blattner, C. (2008). Histories of engagement and sociolinguistic awareness in study abroad: Colloquial French. In L. Ortega & H. Byrnes (Eds.), *The longitudinal study of advanced L2 capacities* (pp. 223–246). Routledge.
<https://doi.org/10.4324/9780203871652>
- Kinginger, C., & Carnine, J. (2019). Language learning at the dinner table: Two case studies of French homestays. *Foreign Language Annals*, 52(4), 850–872.
<https://doi.org/10.1111/flan.12431>
- Knouse, S. M. (2012). The acquisition of dialectal phonemes in a study abroad context: The case of the Castilian theta. *Foreign Language Annals*, 45(4), 512–542.
<https://doi.org/10.1111/j.1944-9720.2013.12003.x>
- Kubota, R. (2016). The social imaginary of study abroad: Complexities and contradictions. *The Language Learning Journal*, 44(3), 347–357.
<https://doi.org/10.1080/09571736.2016.1198098>
- Kurata, N. (2007). Language choice and second language learning opportunities in learners' social networks: A case study of an Australian learner of Japanese. *Australian Review of Applied Linguistics*, 30(1), 5.1–35.18.
<https://doi.org/10.2104/ara10705>
- Kurata, N. (2012). Social networks in second language acquisition. In C. A. Chapelle (Ed.), *The encyclopedia of applied linguistics* (pp. 5248–5256). Blackwell.
<https://doi.org/10.1002/9781405198431.wbeal1080>
- Lafford, B. A., & Uscinski, I. (2014). Study abroad and second language Spanish. In K. Geeslin (Ed.), *The handbook of Spanish second language acquisition* (pp. 386–403). Wiley. <https://doi.org/10.1002/9781118584347.ch22>
- Landau, E. (1895, Oct. 20). Zur relativen Wertbemessung der Turnierresultate. *Deutsches Wochensach*, 11(42), 366–369.
<https://books.google.de/books?id=rDr8AmfYCFkC>
- Lapkin, S., Hart, D., & Swain, M. (1995). A Canadian interprovincial exchange: Evaluating the linguistic impact of a three-month stay in Quebec. In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 67–94). John Benjamins. <https://doi.org/10.1075/sibil.9.06lap>
- Lee, N., Mikesell, L., Joaquin, A. D. L., Mates, A. W., & Schumann, J. (2009). *The interactional instinct: The evolution and acquisition of language*. Oxford University Press. <https://doi.org/10.1093/acprof:oso/9780195384246.001.0001>
- Li, W. (M.) & Gong, Y. (F.). (2022). Agency in the nexus of identity and social network: Understanding the second language socialization experiences of international students in China. *Journal of Language, Identity & Education*, 1–19.
<https://doi.org/10.1080/15348458.2022.2092484>

- Liddon, L., Kingerlee, R., & Barry, J. A. (2018). Gender differences in preferences for psychological treatment, coping strategies, and triggers to help-seeking. *British Journal of Clinical Psychology*, 57(1), 42–58. <https://doi.org/10.1111/bjc.12147>
- Lightbown, P. M., & Spada, N. (2021). *How languages are learned* (5th ed.). Oxford University Press.
- Llanes, À., & Muñoz, C. (2009). A short stay abroad: Does it make a difference? *System*, 37, 353–365. <https://doi.org/10.1016/j.system.2009.03.001>
- Long, M. (1996). The role of the linguistic environment in second language acquisition. In W. Ritchie & T. Bhatia (Eds.), *Handbook of second language acquisition* (pp. 413–468). Academic Press.
- Lowie, W. M., Verspoor, M. H., & de Bot, K. (2009). A dynamic view of second language development across the lifespan. In K. de Bot & R. Schrauf (Eds.), *Language development over the lifespan* (pp. 125–146). Routledge. <https://doi.org/10.4324/9780203880937>
- Mackey, A. (1999). Input, interaction and second language development. *Studies in Second Language Acquisition*, 21(4), 557–587. <https://doi.org/10.1017/S0272263199004027>
- Magnan, S. S., & Back, M. (2007). Social interaction and linguistic gain during study abroad. *Foreign Language Annals*, 40(1), 43–61. <https://doi.org/10.1111/j.1944-9720.2007.tb02853.x>
- Marijuan, S., & Sanz, C. (2018). Expanding boundaries: Current and new directions in study abroad research and practice. *Foreign Language Annals*, 51(1), 185–204. <https://doi.org/10.1111/flan.12323>
- Marriot, H. (1995). The acquisition of politeness patterns by exchange students in Japan. In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 197–224). John Benjamins. <https://doi.org/10.1075/sibil.9.13mar>
- Mayers, A. (2013). *Introduction to statistics and SPSS in psychology*. Pearson Education.
- McManus, K. (2019). Relationships between social networks and language development during study abroad. *Language, Culture and Curriculum*, 32(3), 270–284. <https://doi.org/10.1080/07908318.2019.1661683>
- McManus, K., & Mitchell, R. F. (2015). Subjunctive use and development in L2 French: A longitudinal study. *Language, Interaction and Acquisition*, 6(1), 42–73. <https://doi.org/10.1075/lia.6.1.02mcm>
- McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1), 415–444. <https://doi.org/10.1146/annurev.soc.27.1.415>
- Mikal, J. P., Yang, J., & Lewis, A. (2015). Surfing USA: How Internet use prior to and during study abroad affects Chinese students' stress, integration, and cultural learning while in the United States. *Journal of Studies in International Education*, 19(3), 203–224. <https://doi.org/10.1177/1028315314536990>

- Miller, L., & Ginsberg, R. B. (1995). Folklinguistic theories of language learning. In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 293–315). John Benjamins. <https://doi.org/10.1075/sibil.9.18mil>
- Milroy, J., & Milroy, L. (1978). Belfast: Change and variation in an urban vernacular. In P. Trudgill (Ed.), *Sociolinguistic patterns in British English* (pp. 19–36). Arnold.
- Mitchell, K. (2012). A social tool: Why and how ESOL students use Facebook. *CALICO Journal*, 29(3), 471–493. <https://doi.org/10.11139/cj.29.3.471-493>
- Mitchell, R. (2023). Researching study abroad: Tracking social networks. In C. Pérez-Vidal & C. Sanz (Eds.), *Methods in study abroad research: Past, present, and future* (pp. 345–372). John Benjamins. <https://doi.org/10.1075/rmal.4.14mit>
- Mitchell, R., Myles, F., & Marsden, E. (2019). *Second language learning theories* (4th ed.). Routledge. <https://doi.org/10.4324/9781315617046>
- Mitchell, R., Tracy-Ventura, N., & Huensch, A. (2020). After study abroad: The maintenance of multilingual identity among Anglophone languages graduates. *The Modern Language Journal*, 104(2), 327–344. <https://doi.org/10.1111/modl.12636>
- Mitchell, R., Tracy-Ventura, N., & McManus, K. (Eds.) (2015). *Social interaction, identity and language learning during residence abroad*. European Second Language Association. <http://www.eurosla.org/monographs/EM04/EM04tot.pdf>
- Mitchell, R., Tracy-Ventura, N., & McManus, K. (2017). *Anglophone students abroad: Identity, social relationships and language learning*. Routledge. <https://doi.org/10.4324/9781315194851>
- Mizumoto, A. (2023). Calculating the relative importance of multiple regression predictor variables using dominance analysis and random forests. *Language Learning*, 73(1), 161–196. <https://doi.org/10.1111/lang.12518>
- Mocanu, V., & Llorca, E. (2020). Constructing and reconstructing attitudes towards language learning in study abroad. In A. Bocanegra-Valle (Ed.), *Applied linguistics and knowledge transfer: Employability, internationalisation and social challenges* (pp. 181–202). Peter Lang. <https://doi.org/10.3726/b16992>
- Modirkhamene, S. (2006). The reading achievement of third language versus second language learners of English in relation to the interdependence hypothesis. *International Journal of Multilingualism*, 3(4), 280–295. <https://doi.org/10.2167/ijm043.0>
- Mora, J. C. (2007). Learning context effects on the acquisition of a second language phonology. In C. Pérez-Vidal, M. Juan-Garau, & A. Bel (Eds.), *A portrait of the young in the new multilingual Spain* (pp. 241–263). Multilingual Matters. <https://doi.org/10.21832/9781847690241-015>
- Moreno Bruna, A. M., & Goethals, B. (2020). Intercultural guidance abroad: Impact on social network formation and L2 self-perceived development. *Study Abroad Research in Second Language Acquisition and International Education*, 5(1), 45–68. <https://doi.org/10.1075/sar.18011.bru>
- Newman, M. (2018). *Networks* (2nd ed.). Oxford University Press.

- Ogden, A. (2007). The view from the veranda: Understanding today's colonial student. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 15(1), 35–56. <https://doi.org/10.36366/frontiers.v15i1.215>
- Ortega, L. (2011). SLA after the social turn: Where cognitivism and its alternative stand. In D. Atkinson (Ed.), *Alternative approaches to second language acquisition* (pp. 167–180). Routledge. <https://doi.org/10.4324/9780203830932>
- Ortega, L., & Byrnes, H. (2008). The longitudinal study of advanced L2 capacities: An introduction. In L. Ortega & H. Byrnes (Eds.), *The longitudinal study of advanced L2 capacities* (pp. 3–20). Routledge. <https://doi.org/10.4324/9780203871652>
- Ortega, L., & Iberri-Shea, G. (2005). Longitudinal research in second language acquisition: Recent trends and future directions. *Annual Review of Applied Linguistics*, 25, 26–45. <https://doi.org/10.1017/S0267190505000024>
- Paradowski, M. B. (2017). *M/Other tongues in language acquisition, instruction, and use*. Institute of Applied Linguistics, University of Warsaw.
- Paradowski, M. B., Cierpich-Kozieł, A., Chen, C.-C., & Ochab, J. K. (2022). How output outweighs input and interlocutors matter for study-abroad SLA: Computational Social Network Analysis of learner interactions. *The Modern Language Journal*, 106(4), 694–725. <https://doi.org/10.1111/modl.12811>
- Paradowski, M. B., Jarynowski, A., Czopek, K., & Jelińska, M. (2021a). Peer interactions and second language learning: The contributions of Social Network Analysis in Study Abroad vs At-Home environments. In R. Mitchell & H. Tyne (Eds.), *Language, mobility and study abroad in the contemporary European context* (pp. 99–116). Routledge. <https://doi.org/10.4324/9781003087953-10>
- Paradowski, M. B., Jarynowski, A., Jelińska, M., & Czopek, K. (2021b). Out-of-class peer interactions matter for second language acquisition during short-term overseas sojourns: The contributions of Social Network Analysis. *Language Teaching*, 54(1), 139–143. <https://doi.org/10.1017/S0261444820000580>
- Pellegrino Aveni, V. A. (2005). *Study abroad and second language use: Constructing the self*. Cambridge University Press. <https://doi.org/10.1017/cbo9780511620584>
- Pérez-Vidal, C. (Ed.). (2014). *Language acquisition in study abroad and formal instruction contexts*. John Benjamins. <https://doi.org/10.1075/aals.13>
- Pérez-Vidal, C., & Juan-Garau, M. (2009). The effect of study abroad (SA) on written performance. *EUROSLA Yearbook*, 9(1), 269–295. <https://doi.org/10.1075/eurosla.9.13per>
- Pérez-Vidal, C., & Sanz, C. (2023). *Methods in study abroad research: Past, present, and future*. John Benjamins. <https://doi.org/10.1075/rmal.4>
- Pérez Vidal, C., & Shively, R. L. (2019). L2 pragmatic development in study abroad settings. In N. Taguchi (Ed.), *The Routledge handbook of SLA and pragmatics* (pp. 355–371). Routledge. <https://doi.org/10.4324/9781351164085-23>
- Pozzi, R. (2021). Learner development of a morphosyntactic feature in Argentina: The case of *vos*. *Languages*, 6(4), Article 193. <https://doi.org/10.3390/languages6040193>

- Pozzi, R., & Bayley, R. (2021). The development of a regional phonological feature during a semester abroad in Argentina. *Studies in Second Language Acquisition*, 43(1), 109–132. <https://doi.org/10.1017/S0272263120000303>
- Regan, V. (1995). The acquisition of sociolinguistic native speech norms: Effects of a year abroad on second language learners of French. In B. F. Freed (Ed.), *Second language acquisition in a study abroad context* (pp. 245–267). John Benjamins. <https://doi.org/10.1075/sibil.9>
- Regan, V. (2013). Variation. In J. Herschensohn & M. Young-Scholten (Eds.), *The Cambridge handbook of second language acquisition* (pp. 272–291). Cambridge University Press. <https://doi.org/10.1017/CBO9781139051729>
- Regan, V., Howard, M., & Lemée, I. (2009). *The acquisition of sociolinguistic competence in a study abroad context*. Multilingual Matters. <https://doi.org/10.21832/9781847691583>
- Richardson, J. T. E. (2011). Eta squared and partial eta squared as measures of effect size in educational research. *Educational Research Review*, 6(2), 135–147. <https://doi.org/10.1016/j.edurev.2010.12.001>
- Rientes, B., & Nolan, E.-M. (2014). Understanding friendship and learning networks of international and host students using longitudinal Social Network Analysis. *International Journal of Intercultural Relations*, 41, 165–180. <https://doi.org/10.1016/j.ijintrel.2013.12.003>
- Ring, S. A., Gardner, D., & Dewey, D. P. (2013). Social network development during study abroad in Japan. In K. Kondo-Brown, Y. Saito-Abbott, S. Satsutani, M. Tsutsui, & A. Wehmeyer (Eds.), *New perspectives on Japanese language learning, linguistics, and culture* (pp. 95–122). National Foreign Language Resource Center, University of Hawai'i at Mānoa.
- Sánchez Hernández, A. (2022). Second language pragmatic development in study abroad contexts: An introduction. *Study Abroad Research in Second Language Acquisition and International Education*, 7(1), 2–23. <https://doi.org/10.1075/sar.21035.san>
- Sandel, T. L. (2014). 'Oh, I'm here!': Social media's impact on the cross-cultural adaptation of students studying abroad. *Journal of Intercultural Communication Research*, 43(1), 1–29. <https://doi.org/10.1080/17475759.2013.865662>
- Sanz, C., & Morales-Front, A. (2018). Introduction: Issues in study abroad research and practice. In C. Sanz & A. Morales-Front (Eds.), *The Routledge handbook of study abroad research and practice* (pp. 1–16). Routledge. <https://doi.org/10.4324/9781315639970>
- Sasaki, M. (2004). A multiple-data analysis of the 3.5-year development of EFL student writers. *Language Learning*, 54(3), 525–582. <https://doi.org/10.1111/j.0023-8333.2004.00264.x>
- Sasaki, M. (2007). Effects of study-abroad experiences on EFL writers: A multiple-data analysis. *The Modern Language Journal*, 91(4), 602–620. <https://doi.org/10.1111/j.1540-4781.2007.00625.x>

- Sasaki, M. (2009). Changes in English as a foreign language students' writing over 3.5 years: A sociocognitive account. In R. Manchón (Ed.), *Writing in foreign language contexts: Learning, teaching, and research* (pp. 49–76). Multilingual Matters. <https://doi.org/10.21832/9781847691859-006>
- Sauer, L., & Ellis, R. (2019). The social lives of adolescent study abroad learners and their L2 development. *The Modern Language Journal*, 103(4), 739–762. <https://doi.org/10.1111/modl.12589>
- Savicki, V. (2010). An analysis of the contact types of study abroad students: The peer cohort, the host culture, and the electronic presence of the home culture in relation to readiness and outcomes. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 19(1), 61–86. <https://doi.org/10.36366/frontiers.v19i1.274>
- Schmidt-Rinehart, B. C., & Knight, S. M. (2004). The homestay component of study abroad: Three perspectives. *Foreign Language Annals*, 37(2), 254–262. <https://doi.org/10.1111/j.1944-9720.2004.tb02198.x>
- Segalowitz, N., & Freed, B. (2004). Context, contact, and cognition in oral fluency acquisition: Learning Spanish in at home and study abroad contexts. *Studies in Second Language Acquisition*, 26(2), 173–199. <https://doi.org/10.1017/S0272263104262027>
- Segalowitz, N., Freed, B., Collentine, J., Lafford, B., Lazar, N., & Díaz-Campos, M. (2004). A comparison of Spanish second language acquisition in two different learning contexts: Study abroad and the domestic classroom. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 10(1), 1–18. <https://doi.org/10.36366/frontiers.v10i1.130>
- Serrano, R., Tragant, E., & Llanes, À. (2012). A longitudinal analysis of the effects of one year abroad. *The Canadian Modern Language Review*, 68(2), 138–163. <https://doi.org/10.3138/cmlr.68.2.138>
- Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B., & Ideker, T. (2003). Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13(11), 2498–2504. <https://doi.org/10.1101/gr.1239303>
- Sharma, D., & Dodsworth, R. (2020). Language variation and social networks. *Annual Review of Linguistics*, 6(1), 341–361. <https://doi.org/10.1146/annurev-linguistics-011619-030524>
- Shiri, S. (2013). Learners' attitudes toward regional dialects and destination preferences in study abroad. *Foreign Language Annals*, 46(4), 565–587. <https://doi.org/10.1111/flan.12058>
- Shiri, S. (2015). The homestay in intensive language study abroad: Social networks, language socialization, and developing intercultural competence. *Foreign Language Annals*, 48(1), 5–25. <https://doi.org/10.1111/flan.12127>

- Shively, R. L. (2016). An activity theoretical approach to social interaction during study abroad. *L2 Journal*, 8(2), 51–75. <https://doi.org/10.5070/L28228670>
- Shrum, W., Cheek Jr., N. H., & Hunter, S. MacD. (1988). Friendship in school: Gender and racial homophily. *Sociology of Education*, 61(4), 227–239. <https://doi.org/10.2307/2112441>
- Spada, N., & Lightbown, P. M. (2009). Interaction research in second/foreign language classrooms. In A. Mackey & C. Polio (Eds.), *Multiple perspectives on interaction: Second language research in honor of Susan M. Gass* (pp. 157–175). Routledge. <https://www.taylorfrancis.com/chapters/edit/10.4324/9780203880852-11/interactionresearch-second-foreign-language-classroomsnina-spada-patsy-lightbown>
- Steinwider, S. (2016). EFL learners' post-sojourn perceptions of the effects of study abroad. *Comparative and International Education /Éducation Comparée et Internationale*, 45(2), Article 5. <https://doi.org/10.5206/CIE-ECI.V45I2.9292>
- Strawbridge, T. (2023a). The effect of “at home” network communication, off-site travel, and extracurricular activity on longitudinal social network development in study abroad. *System*, 117, Article 103119. <https://doi.org/10.1016/j.system.2023.103119>
- Strawbridge, T. (2023b). The relationship between social network typology, L2 proficiency growth, and curriculum design in university study abroad. *Studies in Second Language Acquisition*, 45(5), 1131–1161. <https://doi.org/10.1017/S0272263123000049>
- Strobl, C., & Baten, K. (2021). Writing development during study abroad: The role of language contact and social networks. *Study Abroad Research in Second Language Acquisition and International Education*, 6(1), 124–162. <https://doi.org/10.1075/sar.19015.str>
- Swain, M. (1993). The output hypothesis: Just speaking and writing aren't enough. *Canadian Modern Language Review*, 50(1), 158–164. <https://doi.org/10.3138/cmlr.50.1.158>
- Taguchi, N., & Collentine, J. (2018). Language learning in a study-abroad context: Research agenda. *Language Teaching*, 51(4), 553–566. <https://doi.org/10.1017/S0261444818000265>
- Tarone, E. (2007). Sociolinguistic approaches to second language acquisition research—1997–2007. *The Modern Language Journal*, 91(s1), 837–848. <https://doi.org/10.1111/j.1540-4781.2007.00672.x>
- Tarone, E. (2009). A variationist perspective on the interaction approach. In A. Mackey & C. Polio (Eds.), *Multiple perspectives on interaction: Second language research in honor of Susan M. Gass* (pp. 41–56). Routledge. <https://www.taylorfrancis.com/chapters/edit/10.4324/9780203880852-6/variationist-perspective-interaction-approach-elaine-tarone>

- Tarone, E., & Liu, G.-Q. (1995). Situational context, variation, and second language acquisition theory. In G. Cook & B. Seidlhofer (Eds.), *Principle and practice in applied linguistics: Studies in honour of H. G. Widdowson* (pp. 107–124). Oxford University Press.
- Tracy-Ventura, N., Mitchell, R., & McManus, K. (2016). The LANGSNAP longitudinal learner corpus. In M. Alonso-Ramos (Ed.), *Spanish learner corpus research: Current trends and future perspectives* (pp. 117–142). John Benjamins. <https://doi.org/10.1075/scl.78.05tra>
- Tragant, E., & Muñoz, C. (2023). Longitudinal research: The ELLiC project. In E. Tragant & C. Muñoz (Eds.), *Ten years of English learning at school* (pp. 1–27). Palgrave Macmillan. https://doi.org/10.1007/978-3-031-32759-9_1
- Trentman, E. (2013a). Arabic and English during study abroad in Cairo, Egypt: Issues of access and use. *The Modern Language Journal*, 97(2), 457–473. <https://doi.org/10.1111/j.1540-4781.2013.12013.x>
- Trentman, E. (2013b). Imagined communities and language learning during study abroad: Arabic learners in Egypt. *Foreign Language Annals*, 46(4), 545–564. <https://doi.org/10.1111/flan.12054>
- Trentman, E. (2015). Negotiating gender identities and access to social networks during study abroad in Egypt. In R. Mitchell, N. Tracy-Ventura, & K. McManus (Eds.), *Social interaction, identity and language learning during residence abroad* (pp. 263–280). European Second Language Association. <http://www.eurosla.org/monographs/EM04/Trentman.pdf>
- Tullock, B., & Ortega, L. (2017). Fluency and multilingualism in study abroad: Lessons from a scoping review. *System*, 71, 7–21. <https://doi.org/10.1016/j.system.2017.09.019>
- Twombly, S. B. (1995). “Piropos” and friendships: Gender and culture clash in study abroad. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 1(1), 1–27. <https://doi.org/10.36366/frontiers.v1i1.2>
- Vande Berg, M., Connor-Linton, J., & Paige, R. M. (2009). The Georgetown Consortium study: Intervening in student learning abroad. *Frontiers: The Interdisciplinary Journal of Study Abroad*, 18(1), 1–75. <https://doi.org/10.36366/frontiers.v18i1.251>
- Van Mol, C., & Michielsen, J. (2014). The reconstruction of a social network abroad: An analysis of the interaction patterns of Erasmus students. *Mobilities*, 10(3), 423–444. <https://doi.org/10.1080/17450101.2013.874837>
- Wang, C. (2010). Toward a second language socialization perspective: Issues in study abroad research. *Foreign Language Annals*, 43(1), 50–63. <https://doi.org/10.1111/j.1944-9720.2010.01059.x>
- Wang, T., & Saffran, J. R. (2014). Statistical learning of a tonal language: The influence of bilingualism and previous linguistic experience. *Frontiers in Psychology*, 5, Article 953. <https://doi.org/10.3389/fpsyg.2014.00953>

- Waskom, M. L. (2021). seaborn: Statistical data visualization. *Journal of Open Source Software*, 6(60), Article 3021. <https://doi.org/10.21105/joss.03021>
- Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. Cambridge University Press. <https://doi.org/10.1017/cbo9780511815478>
- Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of ‘small-world’ networks. *Nature*, 393, 440–442. <https://doi.org/10.1038/30918>
- Weber, H., Schwenzer, M., & Hillmert, S. (2020). Homophily in the formation and development of learning networks among university students. *Network Science*, 8(4), 469–491. <https://doi.org/10.1017/nws.2020.10>
- Wilkinson, S. (1998). Study abroad from the participants’ perspective: A challenge to common beliefs. *Foreign Language Annals*, 31(1), 23–39. <https://doi.org/10.1111/j.1944-9720.1998.tb01330.x>
- Wright, C. (2018). Effects of time on task in L2 Mandarin Chinese. In C. Sanz & A. Morales-Front (Eds.), *The Routledge handbook of study abroad research and practice* (pp. 166–180). Routledge. <https://doi.org/10.4324/9781315639970-11>
- Wright, C., & Schartner, A. (2013). ‘I can’t ... I won’t?’ International students at the threshold of social interaction. *Journal of Research in International Education*, 12(2), 113–128. <https://doi.org/10.1177/1475240913491055>
- Zalbidea, J., Issa, B., Faretta-Stutenberg, M., & Sanz, C. (2021). Initial proficiency and L2 grammar development during short-term immersion abroad: Conceptual and methodological insights. *Studies in Second Language Acquisition*, 43(2), 239–267. <https://doi.org/10.1017/S0272263120000376>
- Zhang, J.-X., Chen, D.-B., Dong, Q., & Zhao, Z.-D. (2016). Identifying a set of influential spreaders in complex networks. *Scientific Reports*, 6(1), Article 27823. <https://doi.org/10.1038/srep27823>
- Zhou, S., & Mondragón, R. J. (2004). The rich-club phenomenon in the Internet topology. *IEEE Communications Letters*, 8(3), 180–182. <https://doi.org/10.1109/lcomm.2004.823426>
- Zhou, S., & Rose, H. (2023). “Am I really abroad?” The informal language contact and social networks of Chinese foundation students in the UK. *International Review of Applied Linguistics in Language Teaching*. <https://doi.org/10.1515/iral-2022-0042>
- Zuengler, J., & Miller, E. R. (2006). Cognitive and sociocultural perspectives: Two parallel SLA worlds? *TESOL Quarterly*, 40(1), 35–58. <https://doi.org/10.2307/40264510>

Supporting Information

Additional Supporting Information may be found in the online version of this article at the publisher’s website:

Accessible Summary

4.4 Rank-refining Seed Selection Methods for Budget Constrained Influence Maximisation in Multilayer Networks under Linear Threshold Model

Authors: Michał Czuba, Piotr Bródka

Published at: Social Network Analysis and Mining; April 2025

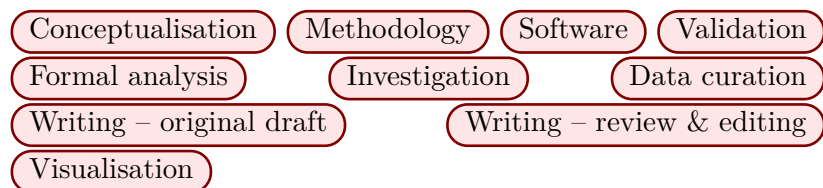
MNiSW points: 100 (IF 2.8)

Citations: Web of Science: 3

(as of 6th March 2026) Scopus: 5

Google Scholar: 8

CRedit:





Rank-refining seed selection methods for budget constrained influence maximisation in multilayer networks under linear threshold model

Michał Czuba¹ · Piotr Bródka¹

Received: 4 November 2024 / Revised: 10 March 2025 / Accepted: 12 March 2025
 © The Author(s) 2025

Abstract

The problem of selecting an optimal seed set to maximise influence in networks has been a subject of intense research in recent years. However, despite numerous works addressing this area, it remains a topic that requires further elaboration. Most often, it is considered within the scope of classically defined graphs with a spreading model in the form of Independent Cascades. In this work, we focus on the problem of budget-constrained influence maximisation in multilayer networks using a Linear Threshold Model. Both the graph model and the spreading process we employ are less prevalent in the literature, even though their application allows for a more precise representation of the opinion dynamics in social networks. This paper aims to answer which of the sixteen evaluated seed selection methods is the most effective and how similar they are. Additionally, we focus our analysis on the impact of spreading model parameters, network characteristics, a budget, and the seed selection methods on the diffusion effectiveness in multilayer networks. Our contribution also includes extending several centrality measures and heuristics to the case of such graphs. The results indicate that all the factors mentioned above collectively contribute to the effectiveness of influence maximisation. Moreover, there is no seed selection method which always provides the best results. However, the seeds chosen with VoteRank-based methods (especially with the $v - \text{rnk} - m$ variant we propose) usually provide the most extensive diffusion.

Keywords Multilayer networks · Linear threshold model · Seed selection · Influence maximisation · Centrality in networks

1 Introduction

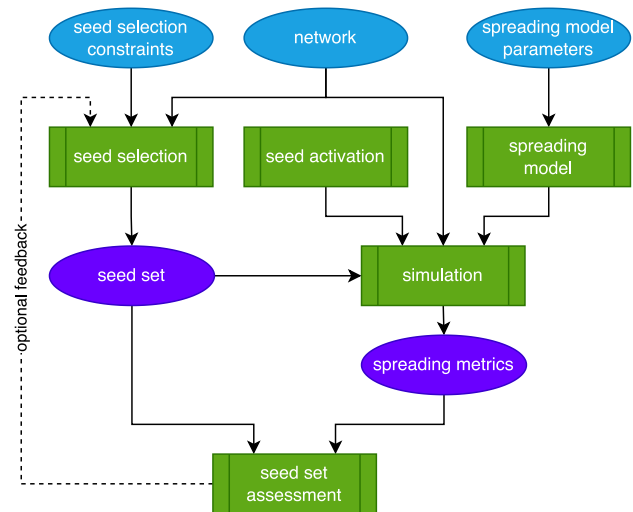
Influence maximisation in social networks aims to identify a set of initial nodes (seeds) that will maximally spread influence throughout the network. Since the initial work of Kempe Kempe et al. (2003) on maximising the spread of influence in social networks, researchers all over the world have been working on inventing and perfecting methods that will produce the best seed set in terms of overall network influence.

The influence maximisation problem depends on several factors (see Fig. 1), which ultimately translate into the dynamics of influence spread in the network and, consequently, the final number of activated agents. We can distinguish here the network itself (e.g. directed, static, temporal, multilayer, hypergraph), the spreading model (e.g. Independent Cascade, Q-Voter, Suspected-Infected-Recovered), its parameters (e.g. β in Suspected-Infected-Recovered), the method of selecting seeds (e.g. random, greedy, degree-based ranking), the way seeds are used (all at once, sequentially during the propagation, etc.), constraints on the seed selection (e.g. budget, cost, time), and property of the influence to be optimised (for instance the final number of activated agents, the number of simulation steps needed to activate 50% of agents in the graph). Moreover, these factors are usually interdependent (as denoted with the feedback loop in Fig. 1). For instance, seed set selection can be closely related to the spreading model, which is very common in methods utilising the optimisation mechanism. As one can note, the number of degrees of freedom of

✉ Michał Czuba
michal.czuba@pwr.edu.pl
 Piotr Bródka
piotr.brodka@pwr.edu.pl

¹ Department of Artificial Intelligence, Wrocław University of Science and Technology, 27 Wybrzeże Wyspiańskiego, 50-370 Wrocław, Poland

Fig. 1 Schematic depiction of the influence maximisation's components. Light blue ellipses — input data, dark blue ellipses — output data, green rectangles — functions, arrows — order of execution



the problem is considerable — the problem of influence maximisation belongs to the NP-hard group Kempe et al. (2003). Furthermore, many of them are interdependent (for instance, the network type and method of selecting seeds). That is why there have been so many studies in the field of influence maximisation, and researchers still address this problem. Specifying all these factors allows for a precise definition of the problem. Below, we refer to them according to the title of our work to show how the problem we tackle is positioned in the literature.

1.1 Seeding process

Research in previous years led to the development of multiple seed selection strategies (methods), which can be divided into three categories: Simulation-based, Heuristic-based, and Mixed Approaches, each differing in their problem-solving perspective.

1.1.1 Seed selection strategy

Simulation-based approaches utilise Monte Carlo (MC) simulations to compute the spread of influence in a network, which can be computationally intensive. These methods perform numerous explicit MC simulations to identify nodes with the highest marginal gain. However, methods belonging to this group may not be computationally efficient for large-scale networks due to the high number of MC simulations required. Researchers have therefore focused on optimising them by either reducing the number of MC simulations needed to find a seed set or the complexity of each MC simulation Singh et al. (2022). The classic greedy algorithm proposed in Kempe et al. (2003) and its extensions such as

CELF Leskovec et al. (2007) or CGA Wang et al. (2010) are examples of methods belonging to this category.

Heuristic-based approaches utilise approximate scoring methods to evaluate the potential influence of each node without relying on time-consuming MC simulations. These methods are often more scalable and efficient compared to simulation-based approaches. Heuristic approaches typically leverage features like network topology or corresponding diffusion model to determine the seed set. They can be categorised into two main groups: rank refinement and model reduction Singh et al. (2022). Rank refinement methods assign a rank to each agent based on computed metrics for estimating the influence spread. The seed set is then determined based on these scores. Examples of such methods include Degree Centrality, Betweenness Centrality, Closeness Centrality, and more sophisticated methods like Single Discount method Sheikhamadi et al. (2015), Degree Punishment method Wang et al. (2016), VoteRank Zhang et al. (2016), community-based methods He et al. (2015), clustering-based methods Bao et al. (2017), network control-based methods Sadaf et al. (2023) and many more. Model reduction methods simplify the computation of expected influence spread by reducing the complexity of the diffusion model. This can be achieved either by reducing stochastic models to deterministic models or by restricting the influence on the local region of the network. Approaches such as Shortest Path Model Kimura and Saito (2006), Local Directed Acyclic Graph heuristic introduced in Chen et al. (2010) or the Independent Path algorithm Kim et al. (2013) are representatives of this category.

According to the taxonomy introduced by Singh et al. (2022), mixed approaches aim to enhance the theoretical efficiency of simulation-based methods while ensuring a guarantee of approximation. These approaches precompute

a set of sketches based on a specific diffusion model to avoid the need for rerunning MC simulations. Subsequently, they leverage these sketches to compute the spread of influence. Mixed approaches were categorised into two classes based on the way the sketches are generated: snapshot-based (e.g. NewGreedy Cheng et al. (2013) or StaticGreedy Zhou et al. (2015)) and Reverse Reachable Sets Borgs et al. (2014), e.g. Sun and Chen (2021). Regarding this kind of division, one can append to this group methods based on artificial intelligence (classic machine learning, deep learning, etc), for instance Tiukhova et al. (2022); Kou et al. (2023); Hajarathaiah et al. (2023); Asgharian Rezaei et al. (2023); Bucur (2020) or Ling et al. (2023).

1.1.2 Seed activation strategy

Another aspect of the seeding process is the seed activation strategy Bródka et al. (2021). Most research focuses on the scenario where all seeds are activated at the same time, but in recent years, researchers started to ask the question, what if we activate only a few seeds at the beginning and see what happens with the spreading process? This led to several approaches like Sequential Seeding Jankowski et al. (2017), Adaptive Seeding Seeman and Singer (2013), Seeding Scheduling Goldenberg et al. (2018) or Active Seeding Sela et al. (2018), which have a different name but the same mechanism behind it. In a nutshell, at the beginning, we select a few seeds using some seed selection strategy. Then, instead of activating them all at the beginning, we activate them in batches, e.g., a few of them at the beginning to start spreading and one more in every iteration, or we add the next batch of seeds when the process is slowing down or dying out. This approach allows us to observe the behaviour of the spreading process and avoid seeding nodes that got activated by their neighbours. Saved seeds can then be used to activate new nodes in sections of the network where no nodes were activated.

1.1.3 Seed selection constraints

The process of selecting seeds is subject to various constraints that must be carefully considered during the problem definition. The most commonly used limitation is a budget, which restricts the number of seeds that can be chosen. In some works, researchers assume this constraint in an edge form — they limit it to one agent Kitsak et al. (2010). Yet, considering only a single super-spreader puts the problem in a different frame. First, the choice of the spreading model gets narrowed down. For example, in LTM, selecting a single-element seed set can result in an inability to trigger diffusion for thresholds that do not oscillate around 0% (and therefore, in an overwhelming number

of cases). On the other hand — such constraint in stochastic diffusion models can make it more difficult to assess such a set as the "spreading potential" of the particular agent depends on many stochastic events (e.g. in ICM), and thus, it can attain a big variance. Another border case is a budget size which leads to an instant saturation of the diffusion. Namely, if one set is too big, then even choosing a seed set randomly will result in a decent diffusion. Thus, if one needs to assess various seed selection methods, the budget size needs to reflect a non-trivial spreading regime. In this work, we follow that approach and do not examine the entire range of budgets. Moreover, we apply different spectra to different diffusion conditions (see Sec. 4). Finally, the effect of this constraint on the rank refining methods is also pronounced. For such approaches, it plays a role of the cutoff applied on the ranking. The bigger it is, the more similar the sets obtained with different methods, as shown later in Fig. 5.

With regard to other forms of constraints, some studies consider the cost associated with each agent Sviridenko (2004). In addition to the limitation of the seed set cardinality, they restrict the overall price of selecting particular agents, which cost differently, akin to a rucksack problem where the objective is to maximise the total influence under a budgetary limit.

Structural network properties, which arise from the network topology, where certain layers or nodes have specific roles or connectivity patterns that influence their suitability as seeds, can also play a role of the constraints imposed on the seed selection process. For instance, in Yuan et al. (2024), authors introduce a seed selection method in the environment consisting of a social network and several information items which spread concurrently and influence each other according to the "association graph" (e.g. spread of item X induces spread of item Y). By that, they solve Multiple Influence Maximisation problem¹ (see Sec. 1.3) constrained by selection of both the agents and the information item to start spread with. Other structural constraints can be rooted in the dynamic nature of the network, where nodes or edges might change over time due to events like link failure, node removal, or structural reconfigurations Michalski et al. (2014). Another important constraint associated with this group is the assumption of network completeness, which predominates in the literature. However, some studies address the challenge of incomplete data Liu et al. (2024).

In addition, we can also imagine attribute-based constraints, which result from the particular form of the graph, i.e. when considering a problem of sexually transmitted diseases that both sexes can carry but affect mostly one of them, e.g. HPV UD (2024).

¹ in their work it is called Multiplex Influence Maximisation.

1.2 Spreading performance metrics

In the influence maximisation problem, the objective function can also adopt many forms. How one defines it can determine the entire experimental setup and strongly affect the conclusions. According to Singh et al. (2022), metrics to measure the performance of the influence maximisation can be divided into four groups: Quality, Efficiency, Scalability, and Robustness. Each of these groups treats the objective differently.

The most commonly occurring is the first one. The basic form of such metric is the number of active agents counted at the steady state of the simulation (see Def. 5). There are also works which define an objective function from a more particular point of view. For instance, the domain of Profit Maximisation addresses the problem of the distinction between the influence and adoption of the advertised product (or opinion), and it extends classical diffusion models to incorporate these factors Lu and Lakshmanan (2012). An exemplary metric, named "profit", that addresses that problem was proposed by Tang et al. (2018), and it combines the benefit obtained from the influence spread and the cost of the influence itself. However, despite being dominant in the literature, metrics belonging to this group have some limitations. Since we focus this study on the influence maximisation efficiency, some of them are addressed in the following parts of the article (see Sec. 2.3).

Works belonging to the second group consider the objective in terms of time. The authors of Singh et al. (2022) mention its representatives, such as Tang et al. (2015). Nonetheless, they allocate to this group only works that measure the time complexity of the influence maximisation algorithm. However, we believe this group can be extended by metrics based on the diffusion time considered as the simulation steps. Here, the fundamental one can be the number of simulation steps in the experiment. Meanwhile, the more sophisticated approaches take the form of a combination of the number of activated agents. For instance, in works from the domain of epidemiology, the important parameter is the maximal growth of the infected population Wątroba and Bródka (2023). Other works consider metrics like the number of simulation steps needed to cover a particular area of the network Sadaf et al. (2023).

The last two groups are Scalability and Robustness. The former one evaluates influence maximisation algorithms not only by their running time but also by memory consumption. Next, Robustness in influence maximisation considers the problem of how an alternation of the diffusion function affects the performance of the seed selection methods. Therefore, a robust method is one which yields

satisfactory results under various diffusion regimes He et al. (2016).

1.3 Influence maximisation in multilayer networks

Despite the wealth of research on finding optimal seed sets or influential nodes (please see the recent review papers Ou et al. (2022); Singh et al. (2022) on that topic), most works are limited to simple one-layer (aka. single-layer) networks. However, in the real world, we are not influenced just by our family, friends, or office colleagues. We are being affected in all social circles. What is more, if we are convinced by our family (to the brand of the smartphone, the idea, the political party, etc.), we will become convinced (active) in all social spheres in which we function and eventually start affecting our acquaintances. In order to capture all those interactions, we need to use the multilayer networks Salehi et al. (2015).

The research on influence maximisation in multilayer networks is much more limited than on one-layer networks. Firstly, there are more degrees of complexity due to the multilayer network structure. Secondly, methods developed for simple one-layer networks often do not work correctly in the multilayer scenario Zhao et al. (2013); Kitsak et al. (2010); Erlandsson et al. (2018). Thus, researchers usually need to start by extending existing methods so they work on multilayer networks and then evaluate them to see which yields the best results. Another problem (that is a reason for the unambiguity in the area of influence maximisation in multilayer networks) is how we define the seed — whether it is a node on a particular layer or an actor represented by various nodes on different layers.

Singh et al. (2022) distinguishes three groups of frameworks that consider multilayer networks: Influence Maximisation Across Multiple Networks (IM2), Multiple Influence Maximisation (MIM), and Multiple Influence Maximisation Across Multiple Networks (MIM2). In this study, we tackle the first of them. It assumes that there is one process that spreads through a multilayer network. The second group mentioned considers that more than one process affects the graph. Although the multilayerance is not introduced directly there, we can depict that problem as a diffusion in the multiplex network, where each layer is independently affected by a different process. Finally, the last group — MIM2, is a superposition of IM2 and MIM, i.e., it tackles the problem of spreading multiple processes in multilayer networks that affect actors, not nodes.

The problem in the assessment of state-of-the-art methods lays in the aforementioned diversity, in which the problem of influence maximisation in multilayer networks is defined. Till now, the experimental research on seed selection for influence spreading in multilayer networks has focused mainly on the Independent Cascade Model (ICM).

In Erlandsson et al. (2018) authors use Degree Centrality, K-shell decomposition Kitsak et al. (2010), VoteRank Zhang et al. (2016) and ARL Erlandsson et al. (2016) on multilayer networks build on Facebook and Twitter interactions and conclude that we can achieve the best results in terms of influence maximisation using Degree Centrality. Another recent study Chen et al. (2020) evaluates Greedy Kempe et al. (2003), out-degree, random, as well as *RRE*, *Fair1* and *Fair2* (proposed in the paper) on four complex networks in the context of Fair Seed Allocation problem.

Another work Wang and Tan (2022a) introduces a multifactorial evolutionary algorithm *MFEA-RIM_m*, leveraging multitasking optimisation theory to address the Robust Influence Maximisation problem in multilayer networks. The same team in another work Wang and Tan (2022b) propose a memetic algorithm named *MA-RIM_{Multi}* to solve the same problem. In their study, they also evaluate (and adapt to the multilayer network case) approaches like *RSPN-3* He et al. (2018), *LAPSO-IM* Singh et al. (2019), *EA-IM* Iacca et al. (2021), *CIMMIC* Xie et al. (2021), or *DGA* Jabari Lotf et al. (2022) — however, their research concerns only ICM.

On the other hand, Lu et al. (2020) describes an application of the Nondominated Sorting Genetic Algorithm II to the problem of Influence Maximization for multilayer networks. The authors of that work compare their method with *LDAG* Chen et al. (2010), Degree Discount Chen et al. (2009), Degree Centrality, Greedy Kempe et al. (2003), and PageRank Page et al. (1999). Interestingly, they use both ICM and Linear Threshold Model (LTM) in their evaluation and utilise the actor-based approach in the spreading model mechanism.

Another influx of works concerning Influence Maximisation in multilayer networks with ICM proposes methods like Clique Influence Maximisation Rao et al. (2022) and K++ Shell Rao and Chowdary (2023). Among methods used in the comparison of these approaches are: Shapley Value-based Influential Nodes Narayanam and Narahari (2011), Knapsack Seeding of Networks Kuhnle et al. (2018), Deep Learning-Based Influence Maximisation Li et al. (2019), Degree Discount Chen et al. (2009), K-Shell Decomposition Carmi et al. (2007), and Community-based K-Shell Decomposition Sun et al. (2021). The same research team introduced the Community-Based Influence Maximisation method Venkatakrishna Rao and Chowdary (2022), which was evaluated on both LTM and ICM and compared the majority of formerly mentioned methods in this paragraph plus approaches like Influence Maximisation on Community Structure Chen and Liu (2020) and *IM-ELPR* Kumar et al. (2021). Nonetheless, the aforementioned team considers the influence maximisation in terms of nodes, which makes

their research closer to the MIM or MIM2 groups than IM2, which we considered in this paper.

Subsequent works concerning LTM and multilayer networks are focused on understanding the model behaviour using just one measure for seed selection Michalski et al. (2013), arbitrarily selecting seeds Zhong et al. (2022), or not selecting seeds at all Zioło et al. (2022).

1.4 Summary and justification for tackling the problem

With regard to the literature review described above, we decided to focus on the problem of seed selection for the LTM in the problem of budget-constrained influence maximisation on multilayer networks. We also limited the evaluated methods to rank-refining heuristics. That allowed us to focus on a particular group and unify the analysis. Moreover, the fact they do not exploit the properties of the spreading model makes them more versatile. They can also be used in works that, instead of LTM, consider its counterparts, like ICM. With that property, they can be treated as a baseline for more sophisticated seed selection methods. Thus, we adjusted the most popular seed selection strategies defined for single-layer networks like the Degree Centrality, PageRank, K-shell Decomposition, etc. (see Sec. 3 for details) as well as the novel ones belonging to the aforementioned group (e.g. Community-Based Influence Maximisation) and evaluated them on twelve multilayer networks varying in size, number of layers and topology (see Tab. 3).

The paper is organised as follows: in Sec. 2, we introduce measures, methods, and models utilised in the research. We also formulate the problem that is being tackled in this paper. Next, in Sec. 3, we describe seed selection methods employed in the study and how we adapted them to multilayer networks. Sec. 4 describes the experimental setup, including evaluated parameters and the list of networks we have used for the evaluation. Finally, we describe the results in Sec. 5 and Sec. 6, describing our step-by-step assessment and findings. The paper ends with the conclusions in Sec. 7.

2 Problem definition

In this section, we define the problem in a formal way, as well as present all structures, definitions and methods that were necessary for the design and execution of the experimental pipeline.

2.1 Multilayer networks

As it was mentioned in Sec. 1, simple one-layer networks might not be enough to understand and model the spread

of influence in the real world. Thus, in this study, we have decided to use multilayer networks Dickison et al. (2016); Kivelä et al. (2014), which allows us to model different interactions in various social cycles. Below, we formally define the network structures used in the following sections.

Definition 2.1 (Network) One can define network as $N = (V, E)$, where V is a non-empty set of nodes and E is a set of edges $(v_1, v_2) : v_1, v_2 \in V$.

Definition 2.2 (Multilayer Network) A tuple $M = (A, L, V, E)$ is called a multilayer network if:

- A is a non-empty set of actors $\{a_1, \dots, a_n\}$,
- L is a non-empty set of layers $\{l_1, \dots, l_m\}$,
- V is a non-empty set of nodes, $V \subseteq A \times L$,
- E is a non-empty set of edges $(v_1^l, v_2^l) : v_1^l, v_2^l \in V$, and $(v_1^{l_1}, v_2^{l_2}) \in E$ then $l_1 = l_2$.

Definition 2.3 (Multiplex Network) Consider a network M , which meets Def. 2. If $\forall l \in L : V_l \equiv A$, then this kind of network is called multiplex.

In other words, a multilayer network can be viewed as a coupled set of networks following Def. 1 so that each actor is represented at most in every layer by the corresponding node ($V \subseteq A \times L$). Multiplex networks only sharpen this property by requiring each actor to be represented in all layers. Moreover, in this work, we do not consider interrelational edges, and links can be produced only within each layer: $(v_1^l, v_2^l) \in E \Rightarrow l_1 = l_2$. For the sake of simplicity, when referring to particular layers, the aforementioned sets will be marked with a subscript. Thus, V_l and E_l refer to the nodes and edges of layer l , respectively.

To provide a visual example of a multilayer network, we present in Fig. 2 a toy representative of such a graph. One can note that it contains:

- eleven actors — $\{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}\}$,
- three layers — $\{l_1, l_2, l_3\}$,
- thirty nodes — $\{v_1^1, v_2^1, v_3^1, \dots, v_1^2, v_2^2, v_3^2, \dots, v_1^3, v_2^3, v_3^3\}$,
- thirty-two edges — $\{(v_1^1, v_4^1), (v_2^1, v_3^1), \dots, (v_1^2, v_2^2), (v_2^2, v_7^2), \dots, (v_1^3, v_4^3), (v_2^3, v_6^3)\}$.

2.2 Linear threshold model

Although the LTM Granovetter (1978) has been comprehensively described in the literature, with its variations and generalisations, we will present its most classical form and its extension to multilayer networks.

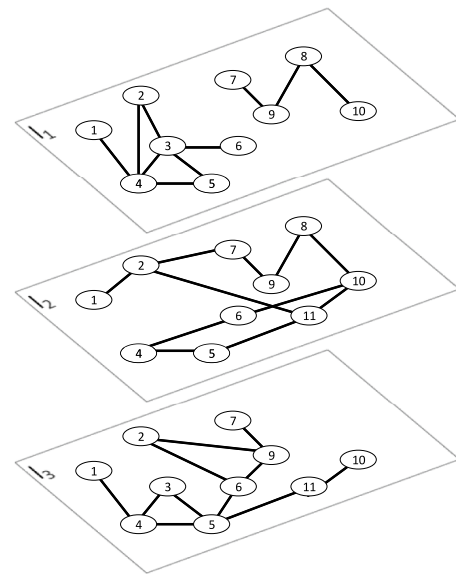


Fig. 2 Example of a multilayer network

Nonetheless, before introducing the models, the auxiliary constructs must be presented:

Definition 2.4 (Binary Discrete System) Let a be an agent² present in the network structure. At time-step t , a can adopt one of two discrete states $x_a(t) \in \{0, 1\}$: *inactive* ($x = 0$), and *active* ($x = 1$) either through interactions with its neighbours or by preserving $x_a(t-1)$. Once $x_a(t) = 1$, a remains active eternally, i.e. there is only one state transition allowed: *inactive* $\rightarrow$ *active*.

Definition 2.5 (Seeds in the Binary Discrete System) Let S_t be a set of active agents in a time step t from the system as defined in Def. 4. We denote S_0 as a seed set, i.e. a set of initially active agents before any dynamics occur in the system. When $S_{t-1} = S_t$ the system reaches a steady state.

2.2.1 Linear threshold model

Following works Granovetter (1978); Kempe et al. (2003), the classical LTM can be formulated as follows:

Definition 2.6 (Linear Threshold Model) Consider a system according to Def. 4 and Def. 5 within a network following Def. 1. Each node v can be influenced by only its neighbours U given connection weights w , so that $\sum_{u \in U} w_{u,v} \leq 1$. Each

² In our case, an agent can be an actor or node depending on the model.

node v chooses a threshold μ_v randomly from the uniform distribution $\mathcal{U}(0, 1)$, which represents a weighted fraction of neighbours of v that must be active in order to activate v . The diffusion process unfolds in discrete steps as follows: a node v at a time step t become active if at $t - 1$ influence of its neighbours U exceeded, μ_v , i.e. $\sum_{u \in U \cap S_{t-1}} w_{u,v} \geq \mu_v$.

2.2.2 Multilayer linear threshold model

The question whether LTM in the form as introduced in Def. 6 can be directly applied to multilayer networks is ambiguous. In the literature, there are two approaches to solving this issue.

The first one proposes to treat each layer separately so that a given actor can be active on some layers and at the same time, be not active on other ones. For instance, works like Rao et al. (2022); Venkatakrishna Rao and Chowdary (2022) adopt this attitude in the problem of influence maximisation. According to Sec. 1.3, this framework can be treated as a variation of MIM. However, we believe that the applicability of that approach is relatively low. Suppose an actor is activated at one layer and still is not at another. In that case, one can make two independent experiments treating each network layer independently — there is no reason to use a multilayer approach. Moreover, in that case, the nodes, not the actors, are the subject of diffusion, so the seed set consists of nodes picked from each layer independently. This approach is nonetheless applicable to problems such as simulating multiple marketing campaigns at the same time. However, if we would like to examine a single spreading process in the multilayer system, it is less useful since decisions made on one layer do not influence decisions on other layers.

The second approach assumes that the actors, not nodes, are the subject of the diffusion process. Therefore, a natural consequence of this condition is to define what it means that an actor is (or is not) active, how it relates to layers (where respective nodes represent it), and where spreading dynamics take place. One of the possible answers to these questions is given in Zhong et al. (2022) by introducing so-called protocols. These are aggregation functions that determine whether to activate an actor by signals from layers where it is represented. In this work, we adopted that approach. Nonetheless, since this study is intended to be an experimental one, we refined the model introduced in Zhong et al. (2022) as follows:

- the heterogeneity (in the sense that each node picks randomly threshold $\mu_v \sim \mathcal{U}(0, 1)$) has been neglected; instead, we impose one value of μ for all actors,
- consequently, heterogeneity of the protocol functions properly for the actors has been transformed into homogeneity over the entire network,

- a requirement for the multiplicity (Def. 3) of the network has been relaxed; thus, any multilayer network can be a medium for the influence propagation model.

The justification for our amendments was an intention to simplify the experimental setup and limit the number of variables, thereby increasing the observability of the impact of μ and protocols on the model performance. Moreover, since this work is purely experimental, lifting the restriction that the network must be a multiplex allows us to test the model on virtually any graph.

Definition 2.7 (Multilayer Linear Threshold Model)

Consider an actor a in the multilayer network $M = (A, L, V, E)$ represented in K layers (i.e. $K \subseteq L$) by a set of its representative nodes: $V_a = \{..., v_a^k, ...\} : k \in K$ and a system according to Def. 4, 5. Each node v can be influenced only by its neighbours U given connection weights w , so that $\forall u \in U : w_{u,v} = \text{degree}(v)^{-1}$ and $\sum_{u \in U} w_{u,v} = 1$. The diffusion process unfolds in discrete steps as follows: node v at time step t receives input y according to threshold μ :

$$y_v(t) = \begin{cases} 1, & \text{if } \sum_{u \in U \cap S_{t-1}} w_{u,v} > \mu \\ 0, & \text{otherwise} \end{cases}$$

Let $y_a(t) = |K|^{-1} \sum_{k \in K} y_v^k(t)$ be the mean input of the actor a in the time step t . Then, given $y_a(t)$, one can determine the state of the actor with a protocol function (Def. 8).

Definition 2.8 (Protocol Function) According to Zhong et al. (2022), state of the actor a of the multilayer network $M = (A, L, V, E)$ in the time step t is determined by a function:

$$x_a(t) = \begin{cases} 1, & \text{if } y_a(t) \geq \delta \text{ or } x_a(t-1) = 1 \\ 0, & \text{otherwise} \end{cases}$$

where $\delta \in [\frac{1}{L}, 1]$ is a parameter.

From the practical point of view, we can notice that, if compared to the classic LTM, the model introduced above (MLTM) adds another threshold by a notion of the protocol. That function behaves like a quantiser that determines whether the actor obtained "sufficient influence" for activation by evaluating the influence obtained by its representative nodes. If so, it will be activated on all layers. Conversely, none of the nodes will be set to active if the mean input is lower than δ . Thus, the activation conditions can be controlled both with μ and δ . It is harder to activate actors for higher values of δ and μ , than if they are lower.

Regarding the above, we decided to examine two extreme cases: $\delta = 1$ (hereinafter called *AND*) and $\delta = \frac{1}{L}$ (hereinafter called *OR*). In the former case, an actor gets activated if it receives sufficient influence on all layers where it is

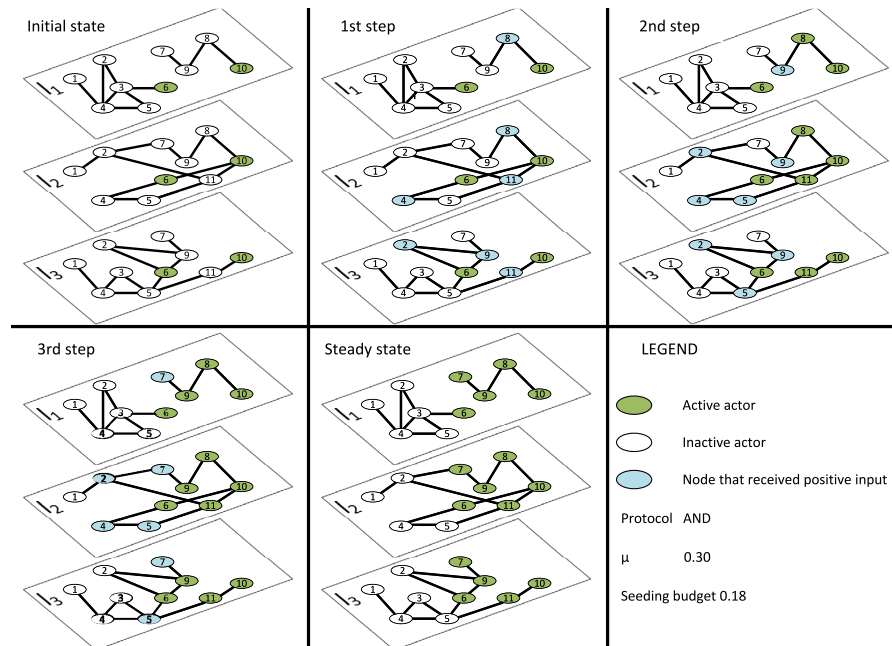


Fig. 3 Example of spreading of MLTM through toy network with protocol *AND*

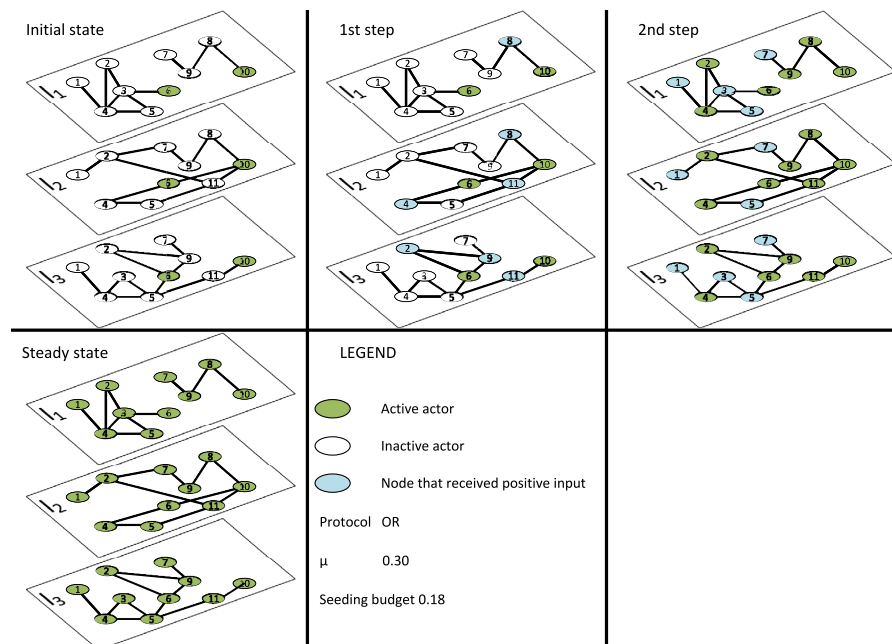


Fig. 4 Example of spreading of MLTM through toy network with protocol *OR*

represented, and conversely, in the latter, where sufficient input in at least one layer is enough for activation. In Figs. 3

and 4, we present differences in these protocols in the example of diffusion under MLTM in the toy network.

Table 1 Indicators of diffusion efficiency taken into account

Symbol	Indicator	Description
DL	Diffusion Length	Number of simulation steps passed until a steady state is reached
G	Gain	Influence under MLTM achieved by a seed set S_0 denoted as a proportion of the actors activated during the diffusion and the total number of activatable actors: $G = 100 \cdot \frac{ S_{ML} - S_0 }{ A - S_0 }$, $G \in [0, 100]$

2.2.3 Influence under MLTM

The issue of measuring the influence under a particular spreading model is thoroughly covered in the literature, as shown in Sec. 1.2. The metrics employed and the properties of the objective function enable a theoretical assessment of the effectiveness of methods designed to solve the influence maximisation problem. Following the seminal work of Kempe et al. (2003), the influence function $\sigma : S \rightarrow \mathbb{N}$ is usually formulated as a function that maps a set of agents into a natural number. This represents the expected number of active agents at the steady state when S is used as the seed set. For the classic LTM, it is proven that σ is submodular, which provides an approximation guarantee of heuristics such as greedy hill climbing Kempe et al. (2003). However, the authors of MLTM do not address this aspect in their work Zhong et al. (2022). In the following section, we aim to resolve this gap.

A function is said to be submodular if it satisfies the diminishing marginal returns property, meaning that the incremental benefit of adding a new element decreases as the set grows. In the context of the considered problem, this implies that the marginal benefit of adding a new actor to an existing seed set diminishes as the seed set becomes larger.

Definition 2.9 (Submodular Function) A function f is submodular if for any sets $S' \subseteq S$ and any element $a \notin S$, the following inequality holds:

$$f(S' \cup \{a\}) - f(S') \geq f(S \cup \{a\}) - f(S)$$

As noted above, the key difference between the classic LTM and its multilayer counterpart lies in the aggregation operation (i.e. a protocol function), where the additional threshold δ is employed. That, however, makes the influence spread non-submodular. We can prove it in the following example. Consider a diffusion within a network from Fig. 2 under MLTM with $\mu = 0.5$ and $\delta = 1/L$ (i.e. variant OR). Then, for a seed set $S_0 = \{a_6, a_{11}\}$, the value of the influence function $\sigma(S_0) = 3$, because $S_{t \rightarrow \infty} = \{a_6, a_{11}, a_{10}\}$. From the other hand, the diffusion for its subset $S'_0 = \{a_{11}\}$, results in $\sigma(S'_0) = 2$, because $S'_{t \rightarrow \infty} = \{a_{11}, a_{10}\}$. However, if these seed sets get extended by the additional actor a_5 , the following occurs: $\sigma(S_0 \cup \{a_5\}) = 11$, because $S \cup \{a_5\}_{t \rightarrow \infty} = \{a_6, a_{11}, a_5, a_4, a_{10}, a_1, a_3, a_2, a_9, a_7, a_8\}$ and

$\sigma(S'_0 \cup \{a_5\}) = 3$, because $S' \cup \{a_5\}_{t \rightarrow \infty} = \{a_5, a_{11}, a_{10}\}$. Finally, we get $\sigma(S' \cup \{a_5\}) - \sigma(S') = 3 - 2 = 1$ and $\sigma(S \cup \{a_5\}) - \sigma(S) = 11 - 3 = 8$ which results in $1 < 8$. Thus, the inequality from Def. 9 is not met.

2.3 Influence maximisation problem

Having proven that the classically formulated influence function under MLTM is not submodular, we are not constrained to use it as a measure of diffusion effectiveness. Therefore, we present below how this paper treats the influence maximisation problem and what metrics are employed to assess the methods used in this study.

Definition 2.10 (Budget Constrained Influence Maximisation under MLTM) Let $\mathcal{S}$ be a family of sets of cardinality s over actors of the multilayer network M in the sense that $\mathcal{S} \subseteq \text{powerset}(N)$. Let $\sigma : \mathcal{S} \rightarrow \mathbb{R}^+$ be an arbitrary function that maps a set of actors to a number denoting the efficiency of using it as a seed set to trigger the diffusion under MLTM. An influence maximisation for seeding budget of size s is a problem of finding $S_0 : \arg \max(\sigma) = S_0 \wedge |S_0| \leq s$.

Following that, we introduce a metric of influence used in this work — Gain (G). Another auxiliary measure (which, however, does not meet Def. 10) is Diffusion Length (DL). It can be used to decide which seed selection methods are more efficient if they exhibit similar G . Table 1 presents their formulas and descriptions.

Nonetheless, it is worth noting that if G is intuitively expected to be high, the value of DL may be open to discussion. In some cases, a great diffusion length is desired. For instance, in viral marketing, an advertiser might prefer to have the message circulating within society for an extended period of time. On the other hand, systems that rely on efficient message passing, such as computer networks, aim to transmit information in the shortest possible time. In this paper, we assume the latter case. Thus, despite the secondary role of DL in evaluating seed selection methods, we consider its lower values to be preferable to higher ones.

3 Seed selection methods used in the study

During the study, we evaluated the following heuristics that can be classified into the rank refined group: Community-Based Influence Maximisation Venkatakrishna Rao and Chowdary (2022), Clique Influence Maximisation Rao et al. (2022), Degree Centrality, Degree Discount Chen et al. (2009), K-shell Decomposition Carmi et al. (2007), K++ Shell Rao and Chowdary (2023), Neighbourhood Size Magnani and Rossi (2011), Neighbourhood Size Discount, PageRank Page et al. (1999), and VoteRank Zhang et al. (2016). We also adopted intuition presented in a work of Kempe et al. (2003) and included in the evaluation greedy hill climbing and random choice as the potential upper and lower boundaries for the other methods (as mentioned above since MLTM is not submodular, we cannot claim that the greedy hill climbing gives us real upper bound approximation Kempe et al. (2003)). Each seed selection method was designed to return a list of all actors from the network, sorted descending by their position in the ranking. Keeping this in mind, one can note that a budget of seeds is a number of the first s actors from the ranking list.

Moreover, due to the specific nature of the problem considered in this study, we had to extend some methods to the case of multilayer networks (i.e. K-shell Decomposition, PageRank, and VoteRank). Thus, we implemented them in two ways. The first one assumed that initial computations are done at the node level and then aggregated for each actor. The second approach took the perspective of the actor from the very beginning. Nonetheless, despite being developed for multilayer networks, some heuristics (e.g. Community-Based Influence Maximisation) addressed the Multiple Influence Maximisation problem, which is not considered in this study. Hence, we also adapted them to the problem described in Sec. 2. For the sake of simplicity, we based experiments on unweighted, undirected networks without self-loops. Therefore, all seed selection methods were defined to work with such graphs (e.g. for Degree Centrality, we did not differentiate out-degree from in-degree).

3.1 Community-based influence maximisation (cbim)

The Community-Based Influence Maximisation proposed by Venkatakrishna Rao and Chowdary (2022) consists of two phases. First, small communities are identified using the function based on Dice Neighbourhood Similarity Dice (1945). Then, they are consolidated to form larger ones

to enhance the community division. In the second phase, an Edge Weight Sum (proposed by the authors of *cbim*) is computed for each node within a community. Finally, nodes are ranked based on these values, and the seed set is selected from the communities using a quota-based approach.

Despite being addressed to multilayer networks, this method finds the most influential nodes, not actors, which does not address the problem considered in this work. Therefore, our amendment was to transform rankings computed for each layer separately by computing a weighted average position (by the layer size) of actors in the rankings and then yield a final rank of the actors by those values. Instead of Edge Weight Sum, we also used a Katz Centrality Katz (1953), which is a similar algorithm in order to speed up the computations.

3.2 Clique influence maximisation (cim)

This method is somehow similar to *cbim*. It was also designed for multilayer networks, and similarly, its objective is to find a seed set of nodes, not actors. The main difference is that instead of dividing the network into communities, it seeks maximal cliques. Then it ranks nodes by their degree and picks iteratively the highest ranked nodes from each clique starting from the largest Rao et al. (2022). For *cim*, our amendment was also to convert rankings computed for each layer to those for actors based on their weighted average position.

3.3 Degree centrality (deg-c)

The next metric we used was the Degree Centrality. It was computed as follows: for each actor a , count nodes connected to the actor a on each layer Magnani and Rossi (2011). Degree Centrality is the number of those nodes.

3.4 Degree centrality discount (deg-c-d)

Degree Centrality Discount approach was initially defined in Chen et al. (2009) as a simple and effective heuristic for ICM. It was based on the concept that highly connected nodes may already have a high level of influence due to their extensive connections, and thus, if they are picked to the seed set, the contribution of their neighbours should be discounted so that seeds are equally distributed through the graph. We have modified it first by neglecting the impact of the spreading model on the discounting process, so that it is based only on the structural properties of the graph. The second modification was using Degree Centrality in the form

defined for multilayer networks Magnani and Rossi (2011), which allowed the method to yield a seed set of the actors.

3.5 Greedy method (*greedy*)

We used a straightforward greedy hill climbing routine, similarly to Kempe et al. (2003). This approach assumes the iterative building of a seed set that results in the best G increment until the budget is spent. We start by evaluating the model's performance with a one-element seed set. We choose an actor a_i that when used, results in the best G . Then, we seek another actor a_j that, used with a_i as seeds, results in the best G or at least the same G but shorter DL . If a few actors result in the same increment of G and the reduction of DL , we choose a random one from them. We repeat this step with the rest of the actors, and as a result, we obtain a greedily chosen seed set.

3.6 K-shell decomposition (*k-sh* and *k-sh-m*)

Nodewise extension (*k-sh*) was designed as follows. For each layer, perform K-shell Decomposition Carmi et al. (2007) and compute the degree of nodes. Then, sort nodes in descending order based on two conditions: the shell they belong to and their degree (respectively, the deeper and the higher, the better). The final ranking list is computed as the weighted average position of an actor in the rankings of layers where it is represented.

The second way we adapted the K-shell Decomposition to multilayer networks utilised the actorwise approach (*k-sh-m*). Compared to the implementation by Carmi et al. (2007), we used the actor's neighbourhood size instead of the node's degree. That follows the intuition of the original algorithm — for a single-layer network, the neighbourhood size of an actor is similar to the degree of its corresponding node. That approach allows graphs to be decomposed into shells that encompass all layers and contain actors, not nodes. Finally, we compute the ranking by sorting actors by the shell to which they were assigned and their degrees.

3.7 K++ shell (*kpp-sh*)

The next method we used was the K++ Shell introduced by Rao and Chowdary (2023) to identify a set of seeds within the multilayer networks. This algorithm, inspired by K-shell, iteratively prunes nodes from the network based on their degree and adds it to shells, but, in addition, introduces reward points for neighbours of nodes pruned in the previous step of the routine. Considering the node's shell number and reward points, the K++ Shell algorithm addresses the limitation of exclusively choosing nodes from the highest bucket. It also utilises a community detection approach to

create the ranking list by picking nodes iteratively from each community according to the seeding budget. However, it operates on nodes similarly to *cbim* and *cim*. We adapted it to the problem considered in the paper similarly by taking the weighted average position (by the layer size) of actors in the rankings for each layer.

3.8 Neighbourhood Size (*nghb-1 s* and *nghb-2 s*)

The Neighbourhood Size of an actor is the number of actors it is connected to, regardless of the connection multiplicity, resulting from layers where interactions between neighbours occur Magnani and Rossi (2011). In experiments, we used two versions of this metric. The first was based on a one-hop neighbourhood (hereinafter *nghb-1 s*) and based on a two-hop setting (*nghb-2 s*). It is worth noting the difference between this metric and Degree Centrality: neighbourhood size does not account for the number of links connecting a pair of actors (which may be repeated due to the network's multilayer structure), whereas this is relevant in the latter method.

3.9 Neighbourhood size discount (*nghb-sd*)

We also decided, based on the Degree Centrality Discount Chen et al. (2009), to propose a Neighbourhood Size Discount by utilising Neighbourhood Size. The method was similar to one described in Sec. 3.4, but instead of the Degree Centrality, we used the one-hop Neighbourhood Size of the actor.

3.10 PageRank (*p-rnk* and *p-rnk-m*)

The node-wise ranking (*p-rnk*) was obtained by computing PageRank, as introduced by its authors in Page et al. (1999), on each network's layer. Then, the final ranking list was created by ordering actors by their mean position in layers rankings.

The second implementation (*p-rnk-m*) adapted the original PageRank algorithm to the case considered in this work differently. Namely, we squeezed a multilayer network into a single-layer graph to preserve all actors. Then, we created edges between actors if at least one layer of the original network had such a link. Then, an output of a PageRank algorithm computed on a squeezed network was a final ranking list.

3.11 Random choice (*random*)

We also examined the random choice of actors. A function `random.random()` from a standard Python library was used to generate a seed set. In order to observe the impact of this selection strategy on diffusion efficiency, the experiment

was repeated 20 times for each evaluated combination of parameters.

3.12 VoteRank (*v-rnk* and *v-rnk-m*)

Our first approach (subsequently *v-rnk*) to adapt VoteRank Zhang et al. (2016) was implemented similarly as in *k-sh*. That is, for each layer, the VoteRank algorithm was executed, resulting in a ranking of nodes that got votes. That list, however, did not contain all nodes, so to satisfy conditions where an extensive seed set was needed, we extended it by nodes that did not get any votes (i.e. were not recognised by the VoteRank as central agents in the network). The final ranking of actors was obtained by computing the weighted mean position of actors in layer-wise ranks.

Another implementation (*v-rnk-m*) was based on a deeply modified VoteRank algorithm so that it could be applied directly to the multilayer networks, i.e. actors, not nodes, are the subject of the algorithm. Therefore, instead of Degree Centrality, we used the Neighbourhood Size as a denominator in the voting potential formula. We also considered connections between actors regardless of layers where that

relation takes place and their multiplicity. Similarly to the original algorithm, our implementation returns the ranking of actors who got at least one vote. Hence, to obtain a complete list, we extended it by the actors that did not obtain any ballots.

3.13 Summary

Having introduced the seed selection methods employed in the study, we show how they work on a toy example. Nonetheless, we refer curious readers to supplementary materials with a source code of their implementation (see Sec. 4.3). Table 2 presents rankings of actors obtained with the aforementioned seed selection methods (without random choice and greedy method, which depend on the spreading model). Please note differences, especially between heuristics, that have been implemented in two ways. As stated earlier, such rankings are being used to construct seed sets. Figure 5 shows how similar they are regarding the budget size. As observed, for smaller budgets, their average similarity does not exceed 30%, but as s increases, similarity grows. This confirms the significant impact of s on different seed selection methods.

Table 2 Positions of each actor in a toy network (Fig. 2) as ranked by the given seed selection method. Green-coloured cells mark the first (dark green), second (light green), and third (olive green) actor in the ranking, respectively

S.S.M.	Actor's ID										
	1	2	3	4	5	6	7	8	9	10	11
cbim	9	5	2	1	7	8	10	3	4	11	6
cim	11	7	3	2	8	4	9	5	6	10	1
deg-c	11	3	5	1	2	6	9	10	4	7	8
deg-c-d	10	4	6	1	2	11	7	8	3	5	9
k-sh	11	3	1	2	5	8	10	6	7	9	4
k-sh-m	11	3	4	1	2	5	9	10	6	7	8
kpp-sh	8	6	3	7	2	10	11	1	4	9	5
nghb-2s	7	1	6	5	8	2	9	11	3	10	4
nghb-1s	9	1	4	3	5	2	10	11	6	7	8
nghb-sd	7	1	8	3	4	2	9	10	5	6	11
p-rnk	11	6	2	3	7	10	9	5	4	8	1
p-rnk-m	11	1	6	3	5	2	10	9	4	7	8
v-rnk	8	6	4	1	7	9	11	5	2	10	3
v-rnk-m	10	1	9	5	6	2	7	8	4	3	11

Fig. 5 Mean pairwise Jaccard similarity with standard deviation between seed sets constructed with the ranking lists from Table 2 in a function of the budget size

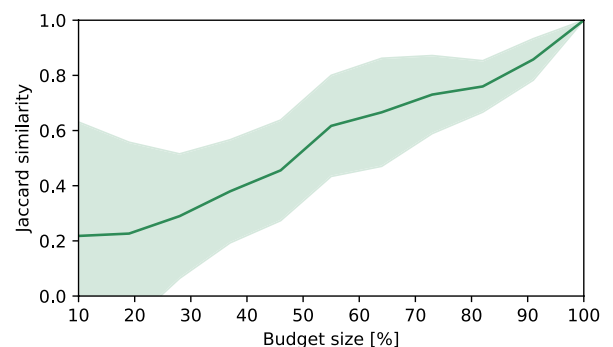


Table 3 Networks used in experiments with their basic parameters shortlisted

Name	Layers	Actors	Nodes	Edges	Note
arxiv	13	14,065	26,796	59,026	Coauthorship network obtained from articles published on the "arXiv" repository De Domenico et al. (2015)
aucs	5	61	224	620	A graph of interactions between employees of Aarhus University, Department of Computer Science Rossi and Magnani (2015)
ckmp	3	241	674	1,370	A network depicting diffusion of innovations among physicians Coleman et al. (1957)
eutr-A	37	417	2,034	3,588	The European air transportation network Cardillo et al. (2013)
lazega	3	71	212	1,659	A network of various types of interactions between staff of a law corporation Snijders et al. (2006)
timik	3	61,702	102,247	881,676	A graph of interactions between users of the virtual world platform for teenagers Jankowski et al. (2017)
er-2	2	1,000	2,000	5,459	Erdős-Rényi artificial network Erdős and Rényi (1960)
er-3	3	1,000	3,000	7,136	Erdős-Rényi artificial network Erdős and Rényi (1960)
er-5	5	1,000	5,000	15,109	Erdős-Rényi artificial network Erdős and Rényi (1960)
sf-2	2	1,000	2,000	4,223	Scale-free artificial network Bollobas et al. (2003)
sf-3	3	1,000	3,000	5,010	Scale-free artificial network Bollobas et al. (2003)
sf-5	5	1,000	5,000	10,181	Scale-free artificial network Bollobas et al. (2003)

is very important. Hence, we decided to use both real and artificially generated graphs. A summary of the networks used in this study, along with a description of the domain context of the data used to construct each network, is provided in Table. 3. All artificial networks were created with the *multinet* library Magnani et al. (2021). The real-world networks were read directly from the raw data, except for the *timik* dataset. Namely, due to its big size, we have limited the network to the interactions from the first quarter of 2009.

The experiments were executed in two stages. During the first one, we wanted to rank seed selection methods preliminarily. Therefore, we utilised smaller networks (i.e. *aucs*, *ckmp*, *eutr-A*, *lazega*, *er-2*, *er-3*, *er-5*, *sf-2*, *sf-3*, *sf-5*) at that phase. Conversely, at the second stage, we were evaluating the five top-so-far heuristics on larger networks (i.e. *arxiv*, *timik*). Ultimately, we used twelve graphs in

the study with sizes up to 880, 000 edges, 100, 000 nodes, 60, 000 actors, and 37 layers.

During the first stage, we have also used single-layer networks to verify the correctness of the extension of some seed selection methods (for instance, *deg-c* and *nghb-s* should yield the same ranking for the single-layer network). However, we decided not to report these results to maintain the brevity of the article. Interested readers can find them in the supplementary materials (see. Sec. 4.3).

4.2 Evaluated parameters of the MLTM

We considered various node activation thresholds (μ), beginning from 0.1 and finishing at 0.9, with the step equals 0.1. It is good to recall that we did not introduce heterogeneity in the model — one value of μ was applied to all actors in the network and, consequently, all nodes. That operation allowed us to discard an additional degree of freedom, which could complicate result analysis.

After the initial series of experiments, we observed that the protocol *AND* is much more demanding than *OR*. That

Table 4 Values of parameters evaluated in the study

Parameter	Evaluated values	
Protocol (δ)	<i>OR</i>	<i>AND</i>
Seed set size (s) [%]	{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30}	{15, 20, 25, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40}
Threshold (μ)	{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9}	
Phase 1 networks	{aucs, ckmp, eutr-A, lazega, er-2, er-3, er-5, sf-2, sf-3, sf-5}	
Phase 2 networks	{arxiv, timik}	
S.S.M	{cbim, cim, deg-c, deg-c-d, greedy ³ , k-sh, k-sh-m, kpp-sh, nghb-2 s, nghb-1 s, nghb-sd, p-rnk, p-rnk-m, random ⁴ , v-rnk, v-rnk-m}	

³ Due to high computational complexity of greedy method we had to limit evaluation only to real networks

⁴ Due to nondeterministic character of the method, each evaluated case was repeated 20 times

is, values of s sufficient to activate the entire network using protocol *OR* were not high enough to trigger spreading when the *AND* strategy was used to aggregate inputs from layers. Therefore, we decided to match different ranges of seeding budgets with each of the protocols (see Table. 4). Moreover, we did not see any sense in experiments with large budgets — that cases are distant from real-world situations. Hence, budget values were chosen to reflect cases where only an actor minority is initially activated.

We also had to determine another factor — how to spend the seed set. Despite the existence of sophisticated mechanisms like sequential seeding (Sec. 1.1.2), we did not use it due to initial assumptions about the fundamental character of this study. Consequently, all actors selected as seeds were activated at the beginning of the simulation.

Finally, each seed selection method had 252 combinations of the MLTM parameters to evaluate on the base networks, plus the top five of them were run on bigger networks. That resulted in 41, 328 designed experiments. All considered parameter ranges are presented in Table. 4.

4.3 Implementation

The experimental environment was implemented in Python and based mainly on *NetworkX* Hagberg et al. (2008) package. A part of the developed code, including a spreading model and seed selection methods, was appended to the *Network Diffusion 0.14.0* Czuba

et al. (2022) library. Nonetheless, we stored the main experimental pipeline and scripts to generate and process results in a separate experimental repository. We also aimed to develop an environment that allows rerunning experiments to reproduce results. Hence, the codebase is published on GitHub (www.github.com/anty-filid/or/rank-refined-seeding-bc-infmax-mlnets-ltm) with manuals describing how to run it. Readers who consider enhancing our study with their own seed selection methods or spreading models are welcome to familiarise themselves with the operating principles of the experimental environment's backbone Czuba et al. (2024). Regarding the computational resources, the experiments were executed on a workstation with Linux Arch 6.5.3. operating system, Intel Xeon Gold 6238 (x86_64 architecture) CPU and 394 GB of RAM. The second machine we used had Windows Server 2012R2 installed and was equipped with Intel Xeon E5-2695 (x86_64 architecture) CPU and 256 GB of RAM.

5 The first stage analysis

As mentioned above, tens of thousands of experiments were conducted during the evaluation. Due to such a large amount of data, finding an appropriate way to process them was necessary. In this chapter, we present step-by-step the methodology applied for processing results, leading to drawing conclusions. Additionally, due to the two-stage

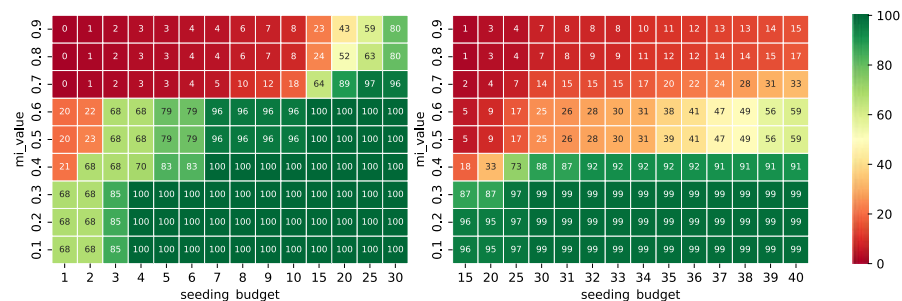
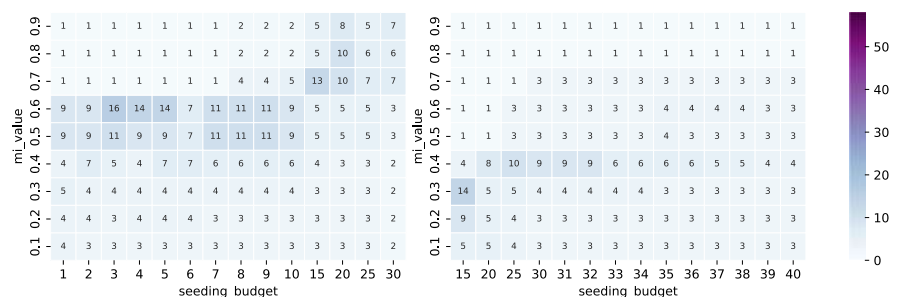


Fig. 6 G achieved by *deg-c-d* seed selection method on *ckmp* network. **Left:** protocol *OR*. **Right:** protocol *AND*

Fig. 7 DL achieved by *deg-c-d* seed selection method on *ckmp* network. **Left:** protocol *OR*. **Right:** protocol *AND*



nature of the experiments, we first present the results obtained for the base networks. Then, after arranging the evaluated methods in a preliminary ranking, we proceeded to the analysis of experiments conducted on large networks and five seed selection methods, which, in the first phase, were ranked highest. Please note that below we present the aggregated outcomes; however, our GitHub repository contains results of every single experiment we have conducted (including plots like in Figs. 6 or 7 for all of them).

5.1 Initial outcomes

The experimental pipeline was triggered separately for each seed selection method. From each of the simulations (i.e. an experiment of the MLTM with particular parameters executed on a particular network), we obtained values of DL and G . They can be easily visualised on heatmaps, as in Figs. 6 and 7.

We analysed them and noticed that their forms differ depending on the applied protocol and seed selection method. However, the most commonly observed pattern was the arrangement of G values into three areas, as in the example image (Fig. 6). The first region ("ineffective") covered a subset of parameters where no diffusion occurred or decayed quickly without affecting the network. The next type ("effective") was opposite to the previous one and encompassed a range of parameters where the process was able to activate all nodes in the network. Finally, there was a transitional zone between them where the process spread was already observable but still far from maximal. It is also worth mentioning the relationship between G and DL heatmaps. The maximum diffusion lengths were typically located in the transitional zone and closely matched that area (see Fig. 7). It was also noticeable that DL reached very similar, low values for effective and ineffective combinations of the μ , s parameters.

These dependencies led us to attempt to empirically estimate functions describing observed areas, which is addressed in Sec. 6.2. Below, we present a bulk description of the obtained G and DL heatmaps for each type of analysed network.

5.1.1 Erdős-Rényi networks

We have noticed that the transitional zone is very narrow for random networks. For the protocol OR , it does not exist at all, and for the protocol AND , it is residual. However, the diffusion length can exceed 50 epochs, which is a substantial value compared to results obtained for different types of graphs. A larger number of layers does not significantly change the diffusion efficiency, although, not surprisingly,

this property helps while protocol OR is used and conversely suppresses diffusion for AND . All evaluated seed selection methods perform similarly in such networks, and even random choice produces satisfactory results (which is expected considering how these networks are constructed).

5.1.2 Scale-free networks

In the case of Scale-free networks, the transitional zone is typically present, particularly for the AND strategy, where it occupies a significant area of the map. While that protocol is applied, the process is not able to cover the entire network, regardless of other model parameters and the seed selection method. For protocol OR , when the number of layers increases, the transitional area gets reduced, and the overall G grows. Also, for more challenging combinations of μ and s , the MLTM is able to cover the entire graph. Finally, for protocol OR , evaluated heuristics are much more efficient than *random*, which is not so noticeable in AND strategy.

5.1.3 Real networks

Real networks revealed the varying effectiveness of the evaluated methods. Furthermore, we observed among them the most significant differences between different implementations of the algorithms (from the perspective of a node and an actor). Therefore, we provide separate descriptions for each of the graphs below.

The transitional region can be clearly observed for the *aucs* graph. For methods that we adapted to the multilayer networks in two ways, both implementations are able to activate the entire graph and yield reasonable results. Generally, the G heatmaps resemble those for the Scale-free networks. Resembling results were obtained for the *ckm-p* graph, although we noticed different behaviour between *k-sh* and *k-sh-m* (in favour of the former). On the other hand, the results for *lazega* network are very similar to Erdős-Rényi networks — there is no transitional zone on heatmaps. In this case, we did not observe any anomalies between different approaches for extending seed selection methods either. We noted significantly different results for the *eutr-A* network. Due to its high density, we achieved a coverage close to the maximal for most of the examined model parameters, regardless of the seed selection method applied. However, a spread triggered with *k-sh*, *kpp-sh*, *random* and *v-rnk* did not reveal this phenomenon, where, for the OR protocol, we obtained a high G value regardless of μ and s . Conversely, for the strategy AND , the diffusion practically did not occur at all. We also noted that *p-rnk* and *p-rnk-m* were somehow complementary to each other for the protocol AND (i.e. for the sets of parameters in which *p-rnk* yielded high G , the *p-rnk-m* was not able to trigger the spread and vice versa).

5.2 Quantitative analysis

Considering the number of obtained partial results, we aggregated them into a form allowing high-level analysis. Namely, for each combination of: protocol, seed selection method, and network, we computed an average G and DL . By that, we could depict results on two-dimensional charts and analyse them independently on the budget size and the activation threshold. Figures 9 and 8 show these heatmaps.

When considering the Fig. 8, it is noteworthy that the average G attained through various seed selection methods exhibits similarity within a specific network. Additionally, correlations between the number of layers and the diffusion efficacy can be observed, as described in Sec. 5.1. The aggregated results affirm that the network type significantly

impacts the achieved G , surpassing the influence of seed selection methods.

Regarding average DL , it is harder to draw valuable conclusions. That comes from the nature of that metric. The final G is the metric that reveals the efficiency of the spread, and the DL resembles only how long the spreading was present in the network. Nonetheless, Fig. 9 shows an average of this value for each network, seed selection method and evaluated protocol of MLTM. Similarly to G , we can conclude that the network type strongly affects the DL for the spread triggered with each seed selection method.

Fig. 8 Mean G achieved by seed selection methods on the base 10 networks. **Left:** protocol *OR*. **Right:** protocol *AND*

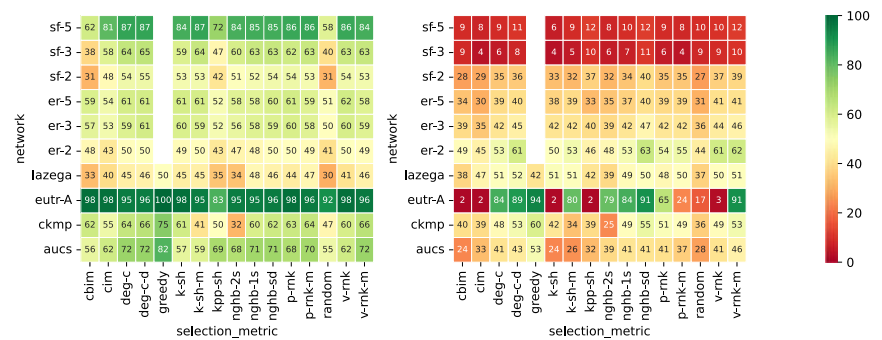


Fig. 9 Mean DL achieved by seed selection methods on the base 10 networks. **Left:** protocol *OR*. **Right:** protocol *AND*

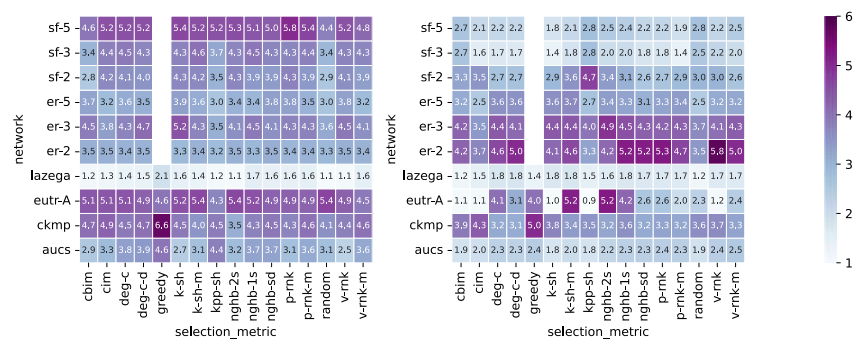
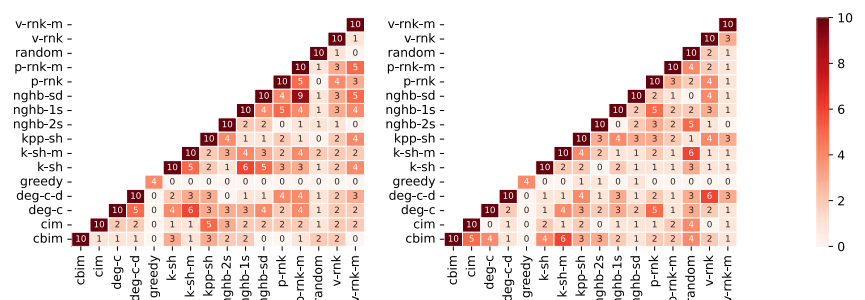


Fig. 10 Occurrences $p > 0.05$ for Wilcoxon tests performed on series $f(\mu, s) > G$ aggregated over the base 10 networks. **Left:** protocol *OR*. **Right:** protocol *AND*



5.3 Statistical analysis

Due to the fact that we obtained comparable outcomes between various seed selection methods regarding the networks used in the experiments, we decided to examine whether the results are statistically similar. Consequently, we employed the Wilcoxon signed rank test for this task. Wilcoxon signed rank test is a non-parametric counterpart of the paired t-test and is often used when one cannot ensure normal distribution of samples Conover (1999); Hassan and Rashid (2020). Since the obtained results consisted of multiple 0 and 100 values of G (i.e. effective and ineffective areas — see Fig. 6), we configured the test to include the zero differences between samples and split them equally between positive and negative ones. The statistics were computed by comparing the aggregated results of G for each seed selection method on each graph to form two data series (one for each protocol). They are depicted on Fig. 10.

Having computed the Wilcoxon test for vectors of G values obtained for seed selection methods on subsequent test cases, one can assess the similarity of how we adapted VoteRank, Page-rank and K-shell Decomposition to multilayer networks. As depicted in Fig. 10, the above methods yielded significantly similar results in less than 50% of comparisons. That is, results for k -sh were similar to k -sh- m in five cases for OR and two for AND . With regard to p -rnk and p -rnk- m , they were similar five times for OR and three times for AND . Finally, the results for v -rnk and v -rnk- m attained similarity only one time for OR , but three times for AND . Hence, it is evident that the adopted variations of these methods are significantly different.

Another interesting conclusion drawn from Fig. 10 can be a similarity of the evaluated seed selection methods to *random* and *greedy*. Here, the protocol also plays an important role. Namely, for OR strategy, *random* differs considerably in most comparisons — it attained similarity to other methods usually one time, probably for the *eutr-A* network, where the spread triggered with all methods attained similarly high G (see Fig. 8). That does not apply to AND , where the Wilcoxon test demonstrated some similarities between *random*, especially for k -sh- m (six times) and *nghb-2 s* (five times). Nonetheless, there are still methods that yield statistically different results than *random* for most networks: *nghb-sd*, *greedy* (no similarity at all); *kpp-sh*, *v-rnk-m* (one time similar); *nghb-1 s*, p -rnk, v -rnk (two times similar). With regard to the similarity of the seed selection methods to *greedy*, it did not occur at all for the strategy OR and, when the second protocol was applied, only for *kpp-sh*, *nghb-2 s* and *nghb-sd* results were similar in one case.

The last thing we report about the Wilcoxon test is the seed selection methods that yielded statistically similar G for most networks. This is important since methods that yield similar results could be used interchangeably depending on

which of them is easier to compute and/or to implement. For protocol OR , p -rnk- m and *nghb-sd* attained similarity for nine networks, while pairs k -sh, *nghb-1 s* and k -sh- m , *deg-c* for six. Furthermore, for AND strategy, the highest number of similarities was six and occurred between pairs: *cbim*, k -sh- m and *deg-c-d*, v -rnk, and k -sh- m , *random*.

5.4 Preliminary rankings

Due to the difficulty in identifying the most effective among the evaluated methods based on average G , we decided to transform the results into a more readable form. To achieve this, we calculated rankings based on the G achieved by the spread using each of the seed selection methods. Namely, for each evaluated combination of the MLTM's parameters and seeding budget on a specific network, we computed the rankings of seed selection methods based on the G obtained for spread triggered in such conditions (the greater G indicated a higher place). In such a way, we received a ranking tensor, which shape was (16, 10, 9, 14, 2) (accordingly, number of seed selection methods, number of networks, number of μ values, number of s values, number of protocols). It was then averaged in the dimensions of μ , s and graph types (real, Scale-free, and Erdős-Rényi). That resulted in final rankings of seed selection methods. Nonetheless, we chose to retain the distinction due to the different propagation natures between OR and AND .

First, in baseline results (Table. 5) for the OR strategy, we can primarily confirm the initial suspicion that the examined methods shall fall between *random* and *greedy*. For the second evaluated protocol, *greedy* seems to be an upper bound, but there are seed selection methods which are classified below *random*. That is for real networks *cbim*, *kpp-sh*, *cim*, k -sh, for Erdős-Rényi graphs *cim*, for Scale-free networks: *cim*, k -sh, and *cbim*. Finally, from the perspective of the bulk average *cim*, *cbim*, and k -sh. That finding reveals how much more difficult the influence maximisation problem is in the AND regime compared to the OR strategy.

It is worth noting that, depending on the protocol, the rankings were shifted in different directions. For the OR strategy, the higher places were usually obtained compared to the AND . The rankings spanned a smaller range of positions for OR strategy (maximal difference for Erdős-Rényi graphs between v -rnk and *random* — 3.28) than for AND (maximal difference for Scale-free networks between *nghb-sd* and *cim* — 7.11). For both cases, these values are not prominent if the total number of the seed selection methods (16 heuristics) is recalled. That indicates that these methods did not significantly stand out from each other.

Having the preliminary rankings, we selected the best methods for evaluation on the large networks. The choice was made based on the "All" columns for both Table. 5 (a) and (b) in a way that we took the top 3 methods, whose union

Table 5 Rankings of the seed selection methods concerning the achieved G for each protocol on the base ten networks, grouped by the graph type. We ignore *greedy* performance in the "All" column because it was computed only for real networks. Greenish cells

(a) Rankings for protocol <i>OR</i> .					(b) Rankings for protocol <i>AND</i> .				
S.S.M.	Networks				S.S.M.	Networks			
	Real	Erdős-Rényi	Scale-free	All		Real	Erdős-Rényi	Scale-free	All
cbim	3.29	3.51	5.44	4.00	cbim	7.23	5.53	8.31	7.04
cim	3.15	4.06	4.81	3.92	cim	6.54	6.75	9.88	7.60
deg-c	2.99	2.61	3.08	2.90	deg-c	3.77	4.88	6.91	5.05
deg-c-d	2.87	2.16	2.80	2.64	deg-c-d	2.62	3.28	4.76	3.46
greedy	1.03	not computed		—	greedy	1.43	not computed		—
k-sh	3.03	2.62	3.98	3.19	k-sh	6.52	5.15	9.62	7.04
k-sh-m	3.56	2.64	3.39	3.23	k-sh-m	5.61	4.95	7.70	6.04
kpp-sh	3.49	3.65	4.06	3.71	kpp-sh	6.95	5.38	6.46	6.33
ngbh-2s	3.86	3.70	4.67	4.05	ngbh-2s	5.04	6.17	7.76	6.19
ngbh-1s	3.16	2.96	3.98	3.35	ngbh-1s	3.53	4.47	6.37	4.66
ngbh-sd	2.97	2.58	3.67	3.06	ngbh-sd	2.58	2.67	2.77	2.66
p-rnk	2.56	2.16	3.22	2.64	p-rnk	3.67	3.82	6.22	4.48
p-rnk-m	2.95	2.46	3.85	3.07	p-rnk-m	4.59	4.22	7.98	5.50
random	4.15	4.86	5.75	4.84	random	6.44	6.40	8.22	6.96
v-rnk	3.01	1.58	3.06	2.59	v-rnk	4.52	3.21	4.98	4.27
v-rnk-m	2.71	3.09	3.75	3.14	v-rnk-m	2.43	3.05	3.34	2.89

produced the 5 elements set. The final set was obtained from the following methods for protocol *OR*: *v-rnk* (score 2.59), *p-rnk* (score 2.64), and *deg-c-d* (score 2.64). Similarly, for *AND* from *ngbh-sd* (score 2.66), *v-rnk-m* (score 2.89), *deg-c-d* (score 3.46). That is *deg-c-d*, *ngbh-sd*, *p-rnk*, *v-rnk*, *v-rnk-m* were chosen for evaluation on the large networks.

6 The second stage analysis

The second stage experiments were conducted on the two largest networks whose properties are described in Table. 3. The larger one, *timik*, has a higher mean degree (2,87) than *arxiv* (0,17). We chose such different networks intentionally to show the efficiency of the seed selection methods while various conditions occur.

6.1 Quantitative analysis

The initial results revealed heatmaps similar to the example depicted in Figs. 6 and 7. However, according to the networks' properties, spread triggered on *timik* yielded much higher G than for *arxiv*. Moreover, for the former graph, the diffusion for the more demanding protocol function — *AND* reached maximal gain much more frequently than for the latter graph. We also observed the longest DL while analysing the initial outcomes for that graph — it reached 95 simulation steps during the experiment with seed selected by *v-rnk* with $s = 15$ and MLTM with $\mu = 0.6$ and the strategy *AND*. Regarding the *arxiv* network, the final efficiency of the spread depended much more on the protocol applied. In the *OR* regime, all evaluated seed selection methods yielded similar results (except the *deg-c-d*, whose effectiveness was lower). Conversely, for protocol *AND*, only seeds determined

Fig. 11 Mean G achieved by seed selection methods on the large networks. **Left:** protocol *OR*. **Right:** protocol *AND*

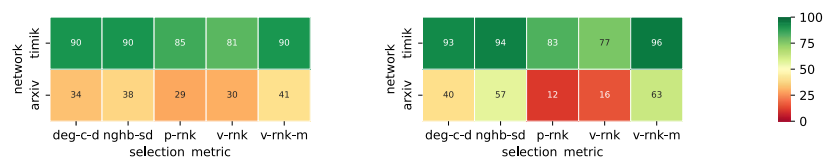
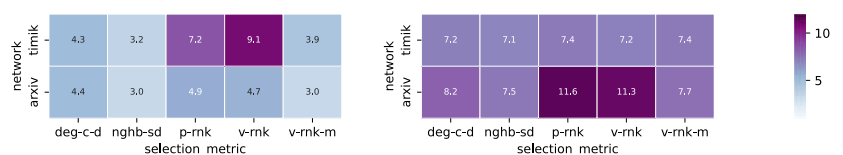


Fig. 12 Mean DL achieved by seed selection methods on the large networks. **Left:** protocol *OR*. **Right:** protocol *AND*



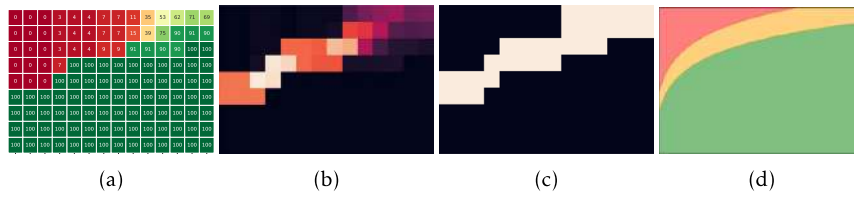


Fig. 13 Example of obtaining efficiency curves for the *v-rnk-m* method on *aucs* network with protocol *OR*. Figure 13 **a** shows a heatmap of the raw gain, **b** computed gradient magnitude, **c** thresholded gradient that divides area into three sections, and **d** the

final curves:: $0.144 \pm 0.013 \cdot \log(s) + 0.331 \pm 0.028$ (for efficient-transitional border), $0.155 \pm 0.020 \cdot \log(s) + 0.431 \pm 0.044$ (for transitional-inefficient border)

Fig. 14 Efficiency curves of the top five seed selection methods computed for *timik* network. **Left:** protocol *OR*. **Right:** protocol *AND*. Mean fitting error for *OR*: $\{\alpha = 11.42\%, \beta = 4.90\%\}$, for *AND*: $\{\alpha = 48.10\%, \beta = 187.40\%\}$

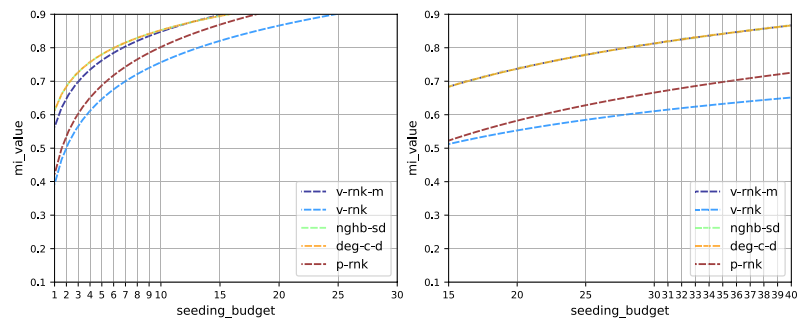
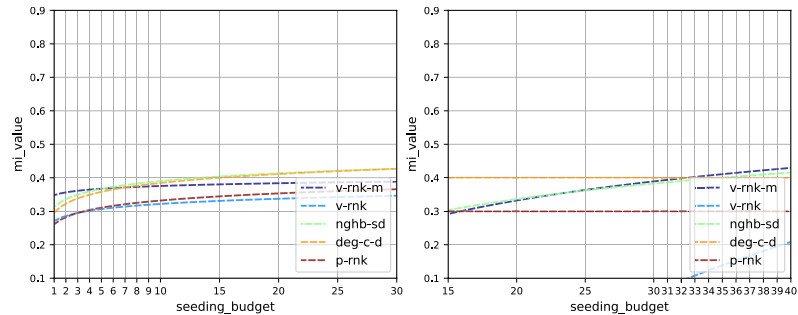


Fig. 15 Efficiency curves of the top five seed selection methods computed for *arxiv* network. **Left:** protocol *OR*. **Right:** protocol *AND*. Mean fitting error for *OR*: $\{\alpha = 46.04\%, \beta = 7.00\%\}$, for *AND*: $\{\alpha = 5.9E14\%, \beta = 7.3E7\%\}$



with *nghb-sd* and *v-rnk-m* caused a high coverage of the graph.

That conclusion can be confirmed while heatmaps of the average *G* and *DL* are plotted (in a similar way as explained in Sec. 5.2). Figures 11 and 12 reveal that *arxiv* network is much more demanding than *timik*. The amplitude of the average *G* is bigger for this graph: it reaches 51 (*AND*, *p-rnk*, *v-rnk-m*), while for *timik* only 19 (*AND*, *v-rnk*, *v-rnk-m*). An inverse relation can be observed for the mean *DL* — the biggest difference occurred for *timik*: 5.9 (*OR* *nghb-sd*, *v-rnk*), which was higher than peak for *arxiv*: 4.1 (*AND* *nghb-sd*, *p-rnk*).

6.2 Efficiency curves

The results obtained during the second stage analyses were also examined for statistical similarities with the Wilcoxon test (as shown in Sec. 5.3). However, none of the pairs constructed from the five seed selection methods demonstrated similarities.

Hence, in order to shed light on the problem from a different perspective, we decided to utilise an observation taken from the raw heatmaps of *G* (see Fig. 6). Namely, we attempted to find functions $\mu = f(s)$ that describe the boundaries between the areas where the diffusion of the MLTM results in heavily different *G*. As denoted in the Sec. 5.1, one can distinguish three regions on such space — effective (where the process results in the *G* reaching

100%), ineffective (where spreading does not take place), and transitional (where DL is the highest, but achieved G is neither big nor close to 0). The idea behind this approach is simple — determining such functions leads to obtaining a general relation between process parameters and G . With them (hereinafter called efficiency curves), one can compare the expected G of the diffusion triggered by various seed selection methods with the MLTM's parameters.

Below, we present a procedure for generating the efficiency curves. At first, we take a raw heatmap depicting relation $f(\mu, s) = G$ (Fig. 13a). Then we compute a gradient: $\nabla f = \left[\frac{\partial G}{\partial s}, \frac{\partial G}{\partial \mu} \right]$ and take its absolute value (Fig. 13b). The next step is an application of the threshold - we use $|\nabla f| > 10$ to filter out small changes of G (Fig. 13c). As a result, we expect to obtain a binary matrix with three areas. If they exist, we can determine points of their borders. Finally, we apply a curve fitting procedure on them to obtain an analytical formula for a function sought (Fig. 13d). We experimented with three of them - a logarithm, a 3rd-degree polynomial and an exponent. The former manifested in the lowest error. Hence, final curves are obtained by fitting $\mu = \alpha \cdot \log(s) + \beta$.

Since the efficient-transitional border is more interesting from the perspective of the influence maximisation problem, we decided to depict only approximations of these functions. Nonetheless, in some cases (especially for protocol *AND*), curve fitting errors were too high to treat them as a reliable approximation of the investigated trend. That usually happened when there was a weak border between regions, and we could not determine it properly with gradient methods. However, we decided to show them all to provide a complete report.

While analysing Fig. 14, one can notice that curves generated for *deg-c-d* and *nghb-s-d* cover each other. That corresponds with Fig. 11, where they are very close concerning the mean G . The same can be observed between *v-rnk-m*, *nghb-s-d*, and *deg-c-d* for the protocol *AND*.

Curves from Fig. 15 for *OR* confirm that *arxiv* network is much more demanding — they are much more flat than their counterparts generated for *timik*. Again, functions obtained for *deg-c-d* and *nghb-s-d* almost cover, which is reflected in Fig. 11. The interesting is a case of *v-rnk-m*, which is below the formerly mentioned curves but still attains higher mean

G as in Fig. 11. Finally, for the *AND* strategy, approximation errors were too big to draw any valuable conclusions.

It is also valuable to compare the growth of curves in terms of μ and s . The first parameter has a more significant impact on the increase in the performance area. This finding can be successfully utilised to solve practical problems. For instance, in a marketing campaign, it is much better to massively expose the audience to presented content (corresponding to a decrease in μ) than to allocate resources to targeted marketing (corresponding to an increase in s). Such an approach allows for much faster coverage of the entire network with the propagated information.

6.3 Final rankings

Similarly to the methodology described in Sec. 5.4, we transformed the raw results into the rankings in order to provide the final assessment of the seed selection methods used in the study. They are presented in Table. 6.

The results reveal a dominance of *v-rnk-m*, the highest classified method for both protocols and on average. The second one was *nghb-s-d*, which also took the second place for all criteria. Finally, the podium was closed by *deg-c-d*, which was ousted by *p-rnk* for *OR* strategy. Table 6 shows another interesting fact. Namely, that the relatively simple heuristics — *nghb-s-d* and *deg-c-d* outperformed much more sophisticated *p-rnk* or *v-rnk*.

However, let us refer to the efficiency curves (Sec. 6.2) while analysing the final ranking. In that case, one can speculate that the critical factor determining the general efficiency of the seed selection method is its behaviour in the demanding environment. In other words, a performance under conditions like parameters determining the transitional area, the *AND* strategy, or if the sparse network is used as a medium for the propagation. Indeed, such circumstances reveal the differences between seed selection methods and make it easier to assess them.

Although this work is not focused on efficient implementations of the employed seed selection methods, we shortly discuss the complexity of those that demonstrate the highest effectiveness, even in their prototype implementations. Table 6 reveals that the most effective methods generally involve more computationally demanding procedures.

Table 6 Rankings of the top five seed selection methods based on the achieved G in the two large networks. Green-shaded cells indicate the first (dark green), second (light green), and third (olive green)-

ranked seed selection methods. The rightmost column presents the computational complexity of the given method

S.S.M.	OR	AND	Avg.	Computational complexity
deg-c-d	3.44	3.27	3.36	$O(A + E + A ^2 L)$
nghb-sd	2.76	2.14	2.45	$O(A + E + A ^2 L)$
p-rnk	2.77	4.13	3.45	$O(N + L (1 + A + E + N) + A \log A)$
v-rnk	2.77	4.24	3.51	$O(N + L (1 + A + E + N \log N + E ^2/ N) + A \log A)$
v-rnk-m	2.13	1.21	1.67	$O(A E + A ^2 + E ^2/ A)$

Before comparing complexities, let us note that in typical multilayer networks, the number of layers (l) is usually small, and connections are sparse, meaning $|N|$ and $|E|$ are of the same order of magnitude. The top-ranked $v-rnk-m$ requires intensive updates to voting scores and iterative selections, leading to a considerably higher computational burden. The two discounting methods, $nhb-sd$ and $deg-c-d$, which ranked second and third, also exhibit a quadratic scaling trend. In contrast, methods based on averaging centralities computed for each actor ($p-rnk$ and $v-rnk$) scale more favourably yet demonstrate the lowest efficiency in terms of performance. These differences highlight an inherent trade-off: while more sophisticated approaches may achieve higher effectiveness, their computational cost remains a critical factor, particularly when scaling to large networks.

7 Conclusions

In this study, we presented an evaluation of various centrality-based seed selection methods for the Budget Constrained Influence Maximisation problem in multilayer networks under the Multilayer Linear Threshold Model.

The first aspect of our work involves proposing the generalisation of three centrality functions (K-shell Decomposition Carmi et al. (2007), PageRank Page et al. (1999), and VoteRank Zhang et al. (2016)) for multilayer networks. They can be further explored in studies focused on centrality in multilayer networks, as these metrics were originally designed to evaluate the centrality of agents (their application to the problem of influence maximisation represents only an alternative way to leverage their capabilities). The next group of methods like Community Based Influence Maximisation Venkatakrisna Rao and Chowdary (2022), Clique Influence Maximisation Rao et al. (2022) and K++ Shell Rao and Chowdary (2023), although originally defined for multilayer networks, were adapted by us to the problem of the influence maximisation based on actors, not nodes. We also introduced a new heuristic — Neighbourhood Size Discount. Finally, all of them (plus a greedy routine and a random choice as hypothetical boundaries) were evaluated.

Our additional contribution is an examination of influence under the MLTM by proving that it is not submodular. We also proposed a new approach to measure it — G , which (in contrast to the classic σ) considers the seed set size.

The next area of our work focuses on assessing the effectiveness of the aforementioned heuristics, both quantitatively and qualitatively. The comprehensive evaluation offered insights into their performance, contributing to understanding their applicability. Apart from G , we developed another way to assess visually the

performance of the seed selection methods in the form of the effectiveness curves. They determine a boundary between the parameters where diffusion effectiveness decreases and where the spread is maximal.

We can conclude that all the evaluated parameters, including the internal parameters of the MLTM, the network type, the budget, and the seed selection method, collectively contribute to the overall effectiveness of influence maximisation. Moreover, there is no seed selection method which always provides the best results. Except for the random choice and greedy approach, which stand out from other results, the mean performance of evaluated methods (Fig. 8) is levelled for protocol *OR*. Interestingly, the same phenomena cannot be observed for more rigorous *AND*, where the mean G is more scattered among seed selection methods. Therefore, in real-world situations (such as opinion propagation), it may sometimes be more effective to focus on reducing the activation threshold (e.g., through mass propaganda or promoting desired attitudes in mass media) rather than searching for a specific set of "influencers". Moreover, when given methods are ranked relatively close to each other, one can consider a criterium of time complexity to choose the proper algorithm so that fast yet akin in results routine is preferred (e.g. lightweight Degree Centrality compared to a heavier PageRank).

The preliminary assessment performed on ten networks revealed that the five methods yielding the best results are Degree Centrality Discount, Neighbourhood Size Discount, PageRank (variant $p-rnk$), and VoteRank (both variants: $v-rnk$, $v-rnk-m$). Finally, the second stage analysis made with the large graphs and heuristics mentioned above surfaced $v-rnk-m$ as the best one. A production-ready implementation of all these methods can be found in the Network Diffusion Czuba et al. (2024) library.

Regarding this study's limitations, it is essential to mention that we analysed only seed selection methods from the rank refined subcategory of the heuristics group (according to Singh et al. (2022)). Moreover, we considered computational complexity only for the most effective methods. However, since all the examined approaches were developed specifically for this work, optimising their implementation for efficiency was not our primary objective. Consequently, the scalability of the employed seed selection methods remains an open avenue for future research and practical applications. Additionally, this work does not explore machine learning approaches recently proposed for single-layer networks. Additionally, we limited the threshold in the spreading model to be homogeneous among actors to limit additional degrees of freedom in the study. Both of these issues can be addressed in future research.

As mentioned, ranking-based seed selection methods do not require knowledge of the propagation dynamics, making their applicability broader. Hence, it could be

interesting to consider them in problems related to influence maximisation and compare to the state-of-the-art methods, like for diffusion source inferring Liu et al. (2025, 2024) or diffusion containment Liu et al. (2024, 2024) which are still fields open to the exploration for multilayer networks. The next interesting topic can be addressing the main limitation of the heuristics we used — a requirement of access to the entire network. That could be done by studies oriented on generalising the machine learning methods for seed selection.

Supplementary information The codebase is published on GitHub (www.github.com/anty-filidor/rank-refined-seeding-bc-infmax-mlnets-ltm) together with manuals describing how to run it, as well as all detailed results for each experiment and simulation we have executed.

Acknowledgements This research was partially supported by the National Science Centre, Poland, under Grant no. 2022/45/B/ST6/04145, the Polish Ministry of Education and Science within the programme “International Projects Co-Funded”, and the EU under the Horizon Europe, grant no. 101086321 OMINO. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the National Science Centre, Polish Ministry of Education and Science, EU or the European Research Executive Agency.

Declarations

Competing interests None.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Asgharian Rezaei A, Munoz J, Jalili M, Khayyam H (2023) A machine learning-based approach for vital node identification in complex networks. *Expert Systems with Applications* 214: 119086 <https://doi.org/10.1016/j.eswa.2022.119086>
- Bao ZK, Liu JG, Zhang HF (2017) Identifying multiple influential spreaders by a heuristic clustering algorithm. *Phys Lett A* 381(11):976–983. <https://doi.org/10.1016/j.physleta.2017.01.043>
- Bollobas B, Borgs C, Chayes J, Riordan O (2003) Directed Scale-Free Graphs, In: Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA) (2003), pp. 132–139. <https://www.microsoft.com/en-us/research/publication/directed-scale-free-graphs/>
- Borgs C, Brautbar M, Chayes J, Lucier B (2014) Maximizing social influence in nearly optimal time. In: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (Society for Industrial and Applied Mathematics, USA, 2014), SODA '14, p. 946–957 <https://doi.org/10.5555/2634074.2634144>
- Bródka P, Jankowski J, Michalski R (2021) Sequential seeding in multilayer networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 31(3), 033130 <https://doi.org/10.1063/5.0023427>
- Bucur D (2020) Top influencers can be identified universally by combining classical centralities. *Sci Rep* 10(1):1–14. <https://doi.org/10.1038/s41598-020-77536-7>
- Cardillo A, Gómez-Gardeñes J, Zanin M, Romance M, Papo D, Pozo Fd, Boccaletti S (2013) Emergence of network features from multiplexity. *Sci Rep* 3(1344):1–6. <https://doi.org/10.1038/srep01344>
- Carmi S, Havlin S, Kirkpatrick S, Shavitt Y, Shir E (2007) A model of internet topology using k -shell decomposition. *Proc Natl Acad Sci* 104(27):11150–11154. <https://doi.org/10.1073/pnas.0701175104>
- Chen J, Liu J (2020) Research on a novel influence maximization algorithm based on community structure. *J Phys: Conf Ser* 1631(1):012064. <https://doi.org/10.1088/1742-6596/1631/1/012064>
- Chen Y, Wang W, Feng J, Lu Y, Gong X (2020) Maximizing multiple influences and fair seed allocation on multilayer social networks. *PLoS ONE* 15(3):e0229201. <https://doi.org/10.1371/journal.pone.0229201>
- Cheng S, Shen H, Huang J, Zhang G, Cheng H (2013) StaticGreedy: solving the scalability-accuracy dilemma in influence maximization. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (Association for Computing Machinery, New York, NY, USA), CIKM '13, pp. 509–518 <https://doi.org/10.1145/2505515.2505541>
- Chen W, Wang Y, Yang S (2009) Efficient influence maximization in social networks, In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Association for Computing Machinery, New York, NY, USA, 2009), KDD '09, p. 199–208 <https://doi.org/10.1145/1555701.15557047>
- Chen W, Yuan Y, Zhang L (2010) Scalable Influence Maximization in Social Networks under the Linear Threshold Model. In: 2010 IEEE International Conference on Data Mining, pp. 88–9 <https://doi.org/10.1109/ICDM.2010.118>
- Coleman J, Katz E, Menzel H (1957) The diffusion of an innovation among physicians. *Sociometry* 20(4):253–270
- Conover WJ (1999) *Practical Nonparametric Statistics* (Wiley, Hoboken, NJ, USA, 1999). <https://www.wiley.com/en-us/Practical+Nonparametric+Statistics%2C+3rd+Edition-p-9780471116067>
- Czuba M, Nurek M, Serwata D, Qi YX, Jia M, Musial K, Michalski R, Bródka P (2024) Network diffusion framework to simulate spreading processes in complex networks. *Big Data Mining Anal* 7(3):637–654. <https://doi.org/10.26599/BDMA.2024.9020010>
- Czuba M, Bródka P (2022) Simulating Spreading of Multiple Interacting Processes in Complex Networks, In: 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA) (IEEE, Shenzhen, China, 2022), pp. 1–10 <https://doi.org/10.1109/DSAA54385.2022.10032425>
- De Domenico M, Lancichinetti A, Arenas A, Rosvall M (2015) Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys Rev X* 5:011027. <https://doi.org/10.1103/PhysRevX.5.011027>

- Dice LR (1945) Measures of the amount of ecologic association. *Between Spec.* <https://doi.org/10.2307/1932409>
- Dickison ME, Magnani M, Rossi L (2016) Multilayer social networks (Cambridge University Press, Cambridge, England, UK, 2016) <https://doi.org/10.1017/CBO9781139941907>
- Erdős P, Rényi A et al (1960) On the evolution of random graphs. *Publ Math Inst Hung Acad Sci* 5(1):17–60
- Erlandsson F, Bródka P, Borg A, Johnson H (2016) Finding influential users in social media using association rule learning. *Entropy* 18(5):164. <https://doi.org/10.3390/e18050164>
- Erlandsson F, Bródka P, Borg A (2018) Seed Selection for Information Cascade in Multilayer Networks. In: *Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications)* (Springer, 2018), pp. 426–436 https://doi.org/10.1007/978-3-319-72150-7_35
- Goldenberg D, Sela A, Shmueli E (2018) Timing matters: Influence maximization in social networks through scheduled seeding. *IEEE Trans Comput Soc Syst* 5(3):621–638. <https://doi.org/10.1109/TCSS.2018.2852742>
- Granovetter M (1978) Threshold models of collective behavior. *Am J Sociol* 83(6):1420–1443
- Hagberg AA, Schult DA, Swart PJ (2008) Exploring Network Structure, Dynamics, and Function using NetworkX. In: *Proceedings of the 7th Python in Science Conference*, ed. by G. Varoquaux, T. Vaught, J. Millman (Pasadena, CA USA, 2008), pp. 11 – 15. <https://aric.hagberg.org/papers/hagberg-2008-exploring.pdf>
- Hajarathiah K, Enduri MK, Dhuli S, Anamalamudi S, Cenkeramaddi LR (2023) Generalization of relative change in a centrality measure to identify vital nodes in complex networks. *IEEE Access* 11: 808–824 <https://doi.org/10.1109/ACCESS.2022.3232288>
- Hassan BA, Rashid TA (2020) Operational framework for recent advances in backtracking search optimisation algorithm: A systematic review and performance evaluation. *Appl Math Comput* 370:124919. <https://doi.org/10.1016/j.amc.2019.124919>
- He JL, Fu Y, Chen DB (2015) A novel top-k strategy for influence maximization in complex networks with community structure. *PLoS ONE* 10(12):e0145283. <https://doi.org/10.1371/journal.pone.0162066>
- He Q, Wang X, Huang M, Lv J, Ma L (2018) Heuristics-based influence maximization for opinion formation in social networks. *Appl Soft Comput* 66:360–369. <https://doi.org/10.1016/j.asoc.2018.02.016>
- He X, Kempe D (2016) Robust Influence Maximization. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York, NY, USA, 2016), KDD '16, p. 885–894 <https://doi.org/10.1145/2939672.2939760>
- Iacca G, Konotopka K, Bucur D, Tonda A (2021) An evolutionary framework for maximizing influence propagation in social networks. *Softw Impacts*. <https://doi.org/10.1016/j.simpa.2021.100107>
- Jabari Lotf J, Abdollahi Azgomi M, Ebrahimi Dishabi MR (2022) An improved influence maximization method for social networks based on genetic algorithm. *Physica A: Statist Mech Appl* 586:126480. <https://doi.org/10.1016/j.physa.2021.126480>
- Jankowski J, Bródka P, Kazienko P, Szymanski BK, Michalski R, Kajdanowicz T (2017) Balancing speed and coverage by sequential seeding in complex networks. *Sci Rep* 7(891):1–11. <https://doi.org/10.1038/s41598-017-00937-8>
- Jankowski J, Michalski R, Bródka P (2017) A multilayer network dataset of interaction and influence spreading in a virtual world. *Sci Data* 4(1):170144. <https://doi.org/10.1038/sdata.2017.144>
- Katz L (1953) A new status index derived from sociometric analysis. *Psychometrika* 18(1):39–43. <https://doi.org/10.1007/BF02289026>
- Kempe D, Kleinberg J, Tardos É (2003) Maximizing the Spread of Influence Through a Social Network. In: *9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146. <https://doi.org/10.1145/956750.956769>
- Kim J, Kim SK, Yu H (2013) Scalable and parallelizable processing of influence maximization for large-scale social networks?. In: *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, pp. 266–277 <https://doi.org/10.1109/ICDE.2013.6544831>
- Kimura M, Saito K (2006) Tractable Models for Information Diffusion in Social Networks. In: *Knowledge Discovery in Databases: PKDD 2006*, ed. by J. Fürnkranz, T. Scheffer, M. Spiliopoulou (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006), pp. 259–27 https://doi.org/10.1007/11871637_27
- Kitsak M, Gallos LK, Havlin S, Liljeros F, Muchnik L, Stanley HE, Makse HA (2010) Identification of influential spreaders in complex networks. *Nat Phys* 6:888–893. <https://doi.org/10.1038/nphys1746>
- Kivela M, Arenas A, Barthelemy M, Gleeson JP, Moreno Y, Porter MA (2014) Multilayer networks. *J Complex Netw* 2(3):203–271. <https://doi.org/10.1093/comnet/cnu016>
- Kou J, Jia P, Liu J, Dai J, Luo H (2023) Identify influential nodes in social networks with graph multi-head attention regression model. *Neurocomputing* 530:23–36. <https://doi.org/10.1016/j.neucom.2023.01.078>
- Kuhnle A, Alim MA, Li X, Zhang H, Thai MT (2018) Multiplex influence maximization in online social networks with heterogeneous diffusion models. *IEEE Trans Comput Soc Syst* 5(2):418–429. <https://doi.org/10.1109/TCSS.2018.2813262>
- Kumar S, Singla L, Jindal K, Grover K, Panda BS (2021) IM-ELPR: Influence maximization in social networks using label propagation based community structure. *Appl Intell* 51(11):7647–7665. <https://doi.org/10.1007/s10489-021-02266-w>
- Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N (2007) Cost-effective outbreak detection in networks. In: *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York, NY, USA, 2007), KDD '07, p. 420–429 <https://doi.org/10.1145/1281192.1281239>
- Ling C, Jiang J, Wang J, Thai MT, Xue L, Song J, Qiu M, Zhao L (2023) Deep graph representation learning and optimization for influence maximization. In: *Proceedings of the 40th International Conference on Machine Learning (JMLR.org, 2023), ICML'23* <https://doi.org/10.5555/3618408.3619288>
- Liu Y, Wang X, Wang X, Yan L, Zhao S, Wang Z (2024) Individual-centralized seeding strategy for influence maximization in information-limited networks. *J R Soc Interface*. <https://doi.org/10.1098/rsif.2023.0625>
- Liu Y, Wang X, Wang X, Wang Z (2024) Fast outbreak sense and effective source inference via minimum observer set. *IEEE/ACM Trans Netw* 32(4):3111–3125. <https://doi.org/10.1109/TNET.2024.3382546>
- Liu Y, Liang G, Wang X, Zhu P, Wang Z (2024) Diffusion containment in complex networks through collective influence of connections. *IEEE Trans Inf Forensics Secur* 19:1510–1524. <https://doi.org/10.1109/TIFS.2023.3338423>
- Liu Y, Zhong Y, Li X, Zhu P, Wang Z (2024) Vital nodes identification via evolutionary algorithm with percolation optimization in complex networks. *IEEE Trans Netw Sci Eng* 11(4):3838–3850. <https://doi.org/10.1109/TNSE.2024.3388994>
- Liu Y, Wang X, Wang X, Wang Z, Kurths J (2025) Diffusion source inference for large-scale complex networks based on network percolation. *IEEE Trans Neural Netw Learning Syst* 36(1):1453–1466. <https://doi.org/10.1109/TNNLS.2023.3321767>

- Li H, Xu M, Bhowmick SS, Sun C, Jiang Z, Cui J (2019) Disco: Influence maximization meets network embedding and deep learning <https://doi.org/10.48550/arXiv.1906.07378>
- Lu Q, Bu Z, Wang Y (2020) A Multiobjective Evolutionary Approach for Influence Maximization in Multilayer Networks, In: *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence* (Association for Computing Machinery, New York, NY, USA, 2020), ICCAI '20, p. 431–438 <https://doi.org/10.1145/3404555.3404568>
- Lu W, Lakshmanan LV (2012) Profit Maximization over Social Networks, In: 2012 IEEE 12th International Conference on Data Mining, pp. 479–488 <https://doi.org/10.1109/ICDM.2012.145>
- Magnani M, Rossi L, Vega D (2021) Analysis of multiplex social networks with r. *J Stat Softw* 98:1–30
- Magnani M, Rossi L (2011) The ML-Model for Multi-layer Social Networks, In: 2011 International Conference on Advances in Social Networks Analysis and Mining (2011), pp. 5–12 <https://doi.org/10.1109/ASONAM.2011.114>
- Michalski R, Kajdanowicz T, Bródka P, Kazienko P (2014) Seed selection for spread of influence in social networks: Temporal vs. static approach. *N Genet Comput* 32(3):213–235. <https://doi.org/10.1007/s00354-014-0402-9>
- Michalski R, Kazienko P, Jankowski J (2013) Convince a dozen more and succeed—the influence in multi-layered social networks, In: 2013 International Conference on Signal-Image Technology & Internet-Based Systems (IEEE, 2013), pp. 499–505 <https://doi.org/10.1109/SITIS.2013.85>
- Narayanan R, Narahari Y (2011) A shapley value-based approach to discover influential nodes in social networks. *IEEE Trans Autom Sci Eng* 8(1):130–147. <https://doi.org/10.1109/TASE.2010.2052042>
- Ou Y, Guo Q, Liu J (2022) Identifying spreading influence nodes for social networks. *Front Eng Manag.* <https://doi.org/10.1007/s42524-022-0190-8>
- Page L, Brin S, Motwani R, Winograd T (1999) The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab. <http://ilpubs.stanford.edu:8090/422/>. Previous number = SIDL-WP-1999-0120
- Rao KV, Chowdary CR (2023) K++ Shell: Influence maximization in multilayer networks using community detection. *Comput Netw* 234:109916. <https://doi.org/10.1016/j.comnet.2023.109916>
- Rao KV, Katukuri M, Jagarapu M (2022) CIM: clique-based heuristic for finding influential nodes in multilayer networks. *Appl Intell* 52(5):5173–5184. <https://doi.org/10.1007/s10489-021-02656-0>
- Rossi L, Magnani M (2015) Towards effective visual analytics on multiplex and multilayer networks. *Chaos, Solitons & Fractals* 72:68–76. <https://doi.org/10.1016/j.chaos.2014.12.022>. **(Multiplex Networks: Structure, Dynamics and Applications)**
- Sadaf A, Mathieson L, Bródka P, Musial K (2023) Maximising influence spread in complex networks by utilising community-based driver nodes as seeds. *Inf Manag Big Data* 25:126–141. https://doi.org/10.1007/978-3-031-35445-8_10
- Salehi M, Sharma R, Marzolla M, Magnani M, Siyari P, Montesi D (2015) Spreading processes in multilayer networks. *IEEE Trans Netw Sci Eng* 2(2):65–83. <https://doi.org/10.1109/TNSE.2015.2425961>
- Seeman L, Singer Y (2013) Adaptive Seeding in Social Networks, In: 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, pp. 459–468 <https://doi.org/10.1109/FOCS.2013.56>
- Sela A, Goldenberg D, Ben-Gal I, Shmueli E (2018) Active viral marketing: incorporating continuous active seeding efforts into the diffusion model. *Expert Syst Appl* 107:45–60. <https://doi.org/10.1016/j.eswa.2018.04.016>
- Sheikhahmadi A, Nematbakhsh MA (2015) Shokrollahi A (2015) Improving detection of influential nodes in complex networks. *Physica A: Stat Mech Appl* 436:833–845. <https://doi.org/10.1016/j.physa.2015.04.035>
- Singh SS, Kumar A, Singh K, Biswas B (2019) Lapso-im: a learning-based influence maximization approach for social networks. *Appl Soft Comput.* <https://doi.org/10.1016/j.asoc.2019.105554>
- Singh SS, Srivastva D, Verma M, Singh J (2022) Influence maximization frameworks, performance, challenges and directions on social network: a theoretical study. *J King Saud Univ-Comput Inf Sci* 34(9):7570–7603. <https://doi.org/10.1016/j.jksuci.2021.08.009>
- Snijders TAB, Pattison PE, Robins GL, Handcock MS (2006) New specifications for exponential random graph models. *Sociol Methodol* 36(1):99–153. <https://doi.org/10.1111/j.1467-9531.2006.00176.x>
- Sun PG, Miao Q, Staab S (2021) Community-based k-shell decomposition for identifying influential spreaders. *Pattern Recogn* 120:108130. <https://doi.org/10.1016/j.patcog.2021.108130>
- Sun G, Chen CC (2021) Influence Maximization Algorithm Based on Reverse Reachable Set. *Mathematical Problems in Engineering* 2021 <https://doi.org/10.1155/2021/5535843>
- Sviridenko M (2004) A note on maximizing a submodular set function subject to a knapsack constraint. *Oper Res Lett* 32(1):41–43. [https://doi.org/10.1016/S0167-6377\(03\)00062-2](https://doi.org/10.1016/S0167-6377(03)00062-2)
- Tang Y, Shi Y, Xiao X (2015) Influence maximization in near-linear time: A martingale approach. In: *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, pp. 1539–1554 <https://doi.org/10.1145/2723372.2723734>
- Tang J, Tang X, Yuan J (2018) Towards Profit Maximization for Online Social Network Providers, In: *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pp. 1178–1186 <https://doi.org/10.1109/INFOCOM.2018.8485975>
- Tiukhova E, Penaloza E, Óskarsdóttir M, Garcia H, Bahnsen AC, Baesens B, Snoeck M, Bravo C (2022) Influencer detection with dynamic graph neural networks <https://doi.org/10.48550/arXiv.2307.08131>
- U.D. of Health & Human Services. HPV Infection (2024). <https://www.cdc.gov/hpv/parents/about-hpv.html>. [Online; accessed 18. May 2024]
- Venkatakrishna Rao K, Chowdary CR (2022) Cbim: Community-based influence maximization in multilayer networks. *Inf Sci* 609:578–594. <https://doi.org/10.1016/j.ins.2022.07.103>
- Wang S, Tan X (2022) Determining seeds with robust influential ability from multi-layer networks: a multi-factorial evolutionary approach. *Knowl-Based Syst* 246:108697. <https://doi.org/10.1016/j.knsys.2022.108697>
- Wang S, Tan X (2022) Solving the robust influence maximization problem on multi-layer networks via a memetic algorithm. *Appl Soft Comput* 121:108750. <https://doi.org/10.1016/j.asoc.2022.108750>
- Wang X, Su Y, Zhao C, Yi D (2016) Effective identification of multiple influential spreaders by degreepunishment. *Physica A* 461:238–247. <https://doi.org/10.1016/j.physa.2016.05.020>
- Wang Y, Cong G, Song G, Xie K (2010) Community-based greedy algorithm for mining top-K influential nodes in mobile social networks. In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York, NY, USA, 2010), KDD '10, p. 1039–1048 <https://doi.org/10.1145/1835804.1835935>
- Wątroba P, Bródka P (2023) Influence of information blocking on the spread of virus in multilayer networks. *Entropy* 25(2):231. <https://doi.org/10.3390/e25020231>
- Xie X, Li J, Sheng Y, Wang W, Yang W (2021) Competitive influence maximization considering inactive nodes and community

- homophily. *Knowl-Based Syst* 233:107497. <https://doi.org/10.1016/j.knosys.2021.107497>
- Yuan Z, Shao M, Chen Z (2024) Graph bayesian optimization for multiplex influence maximization. *AAAI* 38(20):22475–22483. <https://doi.org/10.1609/aaai.v38i20.30255>
- Zhang JX, Chen DB, Dong Q, Zhao ZD (2016) Identifying a set of influential spreaders in complex networks. *Sci Rep* 6(1):27823. <https://doi.org/10.1038/srep27823>
- Zhao D, Li L, Li S, Huo Y, Yang Y (2013) Identifying influential spreaders in interconnected networks. *Phys Scr* 89(1):015203. <https://doi.org/10.1088/0031-8949/89/01/015203>
- Zhong YD, Srivastava V, Leonard NE (2022) Influence spread in the heterogeneous multiplex linear threshold model. *IEEE Trans Control Netw Syst* 9(3):1080–1091. <https://doi.org/10.1109/TCNS.2021.3088782>
- Zhou C, Zhang P, Zang W, Guo L (2015) On the upper bounds of spread for greedy algorithms in social network influence maximization. *IEEE Trans Knowl Data Eng* 27(10):2770–2783. <https://doi.org/10.1109/TKDE.2015.2419659>
- Zioło M, Bródka P, Spoz A, Jankowski J (2022) Modeling the impact of external influence on green behaviour spreading in multilayer financial networks. 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA) pp. 1–10 <https://doi.org/10.1109/DSAA54385.2022.10032397>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

4.5 Applicability of the Minimal Dominating Set for Influence Maximization in Multilayer Networks

Authors: Michał Czuba, Mingshan Jia, Piotr Bródka, Katarzyna Musiał

Published at: Journal of Complex Networks; December 2025

MNiSW points: 100 (IF 1.5)

Citations: Web of Science: 0

(as of 6th March 2026) Scopus: 0

Google Scholar: 0

CRedit:

Conceptualisation Methodology Software Validation
Formal analysis Investigation Data curation
Writing – original draft Writing – review & editing
Visualisation

Journal of Complex Networks, 2025, 6, cnaf036
<https://doi.org/10.1093/comnet/cnaf036>
 Advance access publication 8 December 2025

Article



Applicability of the minimal dominating set for influence maximization in multilayer networks

Michał Czuba ^{1,2,*}, Mingshan Jia ², Piotr Bródka ^{1,2}, Katarzyna Musiał ²

¹Department of Artificial Intelligence, Wrocław University of Science and Technology, 27 Wybrzeże Wyspiańskiego St., 50-370 Wrocław, Lower Silesia, Poland

²Data Science Institute, University of Technology Sydney, 15 Broadway, Ultimo, NSW 2007, Australia

*Corresponding author. Department of Artificial Intelligence, Wrocław University of Science and Technology, 27 Wybrzeże Wyspiańskiego St., 50-370 Wrocław, Lower Silesia, Poland. E-mail: michal.czuba@pwr.edu.pl

ABSTRACT

The minimal dominating set (MDS) is a well-established concept in network controllability and has been successfully applied in various domains, including sensor placement, network resilience, and epidemic containment. In this study, we adapt the local-improvement MDS routine and explore its potential for enhancing seed selection for influence maximization in multilayer networks. We employ the Linear Threshold Model, which provides an intuitive representation of influence spread or opinion dynamics by accounting for the accumulation of peer influence. To ensure interpretability, we utilize rank-refining seed selection methods, with the results further filtered with MDS. Our findings reveal that incorporating MDS into the seed selection process improves spread only within a specific range of situations. Notably, the improvement is observed for larger seed set budgets, lower activation thresholds, and when an AND strategy is used to aggregate influence across network layers. This scenario reflects situations where an individual does not require the majority of their acquaintances to hold a target opinion, but must be influenced across all social circles.

KEYWORDS: minimal dominating set; multilayer networks; influence maximization; linear threshold model; network control.

1. INTRODUCTION

Traditionally, spreading processes, including influence diffusion, were studied on simple, undirected, and unlabelled networks. However, these models fail to represent real-world relationships realistically. With time, more and more complex graph representations were developed, and eventually, a multilayer network concept was proposed. It offers a more nuanced depiction of social interactions than classical graphs. This model captures diverse types of connections—such as communication, professional, family, or social ties, by distinguishing between different layers (i.e. types of relationships) [1, 2]. Hence, incorporating such networks into research on diffusion increases the accuracy of modelling.

One example of spreading processes is influence spread, a key challenge in social network analysis with applications in areas such as viral marketing, information diffusion, and public health interventions [3]. The process is often analyzed through cascade dynamics using mechanisms such

Received: 13 March 2025; **Accepted:** 10 September 2025

© The Author(s) 2025. Published by Oxford University Press. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

as the Linear Threshold Model (LTM) and the Independent Cascade Model (ICM). LTM employs a threshold-based activation rule, where a node becomes active only if the total weight of its active neighbours exceeds a predefined threshold [4, 5]. In contrast, ICM activates nodes probabilistically through independent trials along each edge [6]. While ICM has been extended to multilayer settings [7–10], its limited flexibility in capturing heterogeneous link influences makes LTM a more suitable choice for modelling opinion dynamics in a multilayer environment [11, 12]. Researchers developed various LTM extensions for multilayer networks, e.g. the approach presented in [11] incorporates content-dependent parameters to differentiate link types, while another one [12] introduces layer-specific activation thresholds and heterogeneous activation rules. Given a cascade model, the next key step of influence maximization is to identify seed agents that achieve the widest spread with minimal resources [13]. Numerous methods have been proposed for seed selection in multilayer networks and, as categorized in [14], can be grouped into heuristic techniques [15], simulation-based approaches [16], and hybrid methods incorporating machine learning [17, 18].

Influence maximization can be viewed as a softer version of control, which motivates this study's exploration of incorporating concepts from control theory to assess their applicability in the realm of influence spread. Control theory in network science focuses on strategies to transition a system from one state to another within a defined number of steps [19]. To achieve this, control methods have been developed that identify a minimal set of driver nodes capable of steering the network towards the desired state [20]. For example, a study by [21] discusses the identification of a minimal set of driver nodes for structural controllability in large-scale dynamical systems. Some researchers have examined the impact of multilayer networks on the controllability of these structures [22]. However, real social networks often do not exhibit deterministic behaviours, making strict network control difficult.

In this paper, we aim to bridge concepts from control theory with influence maximization techniques, which have been largely independent of each other. We address this through an exploratory study that employs the Minimal Dominating Set (MDS) in the seed selection process for multilayer networks. We explore the viability and potential advantages of using MDS to maximize influence. Previous research has shown that MDS is an effective seed selection method in single-layer networks [23, 24], but its performance in multilayer networks remains largely unexplored. Note that the primary goal of this study is not to propose a new seed selection method but to assess the effectiveness and usefulness of MDS in improving existing methods.

Our study demonstrates that incorporating MDS into the seed selection process enhances diffusion, but this improvement occurs only within a specific range of spreading parameters. The benefit is most noticeable when the activation thresholds are lower, the AND strategy is used to aggregate influence across network layers, and the seed set budget is larger. Additionally, the enhancement is particularly pronounced in networks resembling scale-free graphs. This scenario is especially relevant when an individual needs to be influenced across all social circles without requiring the majority of their acquaintances to share the target opinion. An example might be a presidential election with highly polarized candidates; we may not know the opinions of all our friends and family because not everyone wants to share them (for instance, they are afraid of what others would think), or we prefer not to discuss politics. However, we usually know the opinions of a few people in each of our social circles, and if the opinions of all (or most) of them are the same, we are more willing to adopt it as our view.

The article is organized as follows. In Section 2, we present a formal definition of MDS, review state-of-the-art methods for obtaining it, and introduce our adaptation of the local-improvement algorithm to the context of multilayer networks. Section 3 outlines the experimental setup used in the study, including metrics and the evaluated parameter space. The results of the analysis are discussed in Section 4, where we examine the properties of the MDSs obtained for the examined graphs, the similarity of seed sets drawn with and without incorporating MDS in the seed selection process, and the effectiveness of that in maximizing influence with a particular attention to scale-free networks. Section 5 concludes the work.

2. MDS IN MULTILAYER NETWORKS

MDS is the minimal set of ‘driver agents’ that are capable of controlling the entire network. The problem of identifying it is well known to be NP-hard [25], and numerous approaches have been proposed to solve it efficiently. In this section, we introduce this problem, provide a review of related works, and introduce an extension of the local improvement algorithm to the multilayer setting.

2.1. Multilayer networks

First, we clarify how we conceptualize multilayer networks. Specifically, we adopt the framework introduced by [1, 2]. To ensure consistency in notation, we restate it as follows:

THEOREM 2.1 (Multilayer Network). A multilayer network can be described as a tuple $M = (A, L, V, E)$ consisting of the following sets:

- actors $A = \{a_1, a_2, \dots\}$,
- layers $L = \{l_1, l_2, \dots\}$,
- nodes $V = \{v_1^1, v_1^2, \dots, v_2^1, v_2^2, \dots\} : V \subseteq A \times L$,
- edges $E = \{(v_1^1, v_2^1), \dots, (v_1^2, v_2^2), \dots\} : (v_1^1, v_2^2) \notin E \wedge (v_1^1, v_2^1) \equiv (v_2^1, v_1^1)$.

In other words, a multilayer network is a coupled set of single-layer networks, where each actor is represented in at most every layer by a corresponding node ($V \subseteq A \times L$), e.g. the representation of actor a_1 in layer l_2 is denoted as v_1^2 . Another property of this model is that interlayer edges are not allowed, meaning links can only exist within the same layer: $(v_1^1, v_2^2) \notin E$. Additionally, to simplify the analysis, edges are considered undirected: $(v_1^1, v_2^1) \equiv (v_2^1, v_1^1)$ and unweighted.

2.2. Dominating sets

Having introduced the network model utilized in this work, we are able to provide the necessary definitions of dominating sets, extending the foundational concepts introduced in [26] to the multilayer scenario.

THEOREM 2.2 (Dominating Set). A set of actors D' dominates the multilayer network M if for every actor $a \in A$, either a belongs to D' , or for each layer where a is represented, the corresponding node of a is adjacent to a node that represents an actor in D' .

THEOREM 2.3 (Minimum Dominating Set). Let $\mathcal{D}$ be a family of dominating sets of the multilayer network M : $\mathcal{D} \subseteq \text{powerset}(A)$. A Minimum Dominating Set is a set $D : D \in \mathcal{D}$, and $|D| = \arg \min_{D' \in \mathcal{D}} |D'|$, i.e. D is the smallest dominating set among all dominating sets in $\mathcal{D}$.

THEOREM 2.4 (Minimal Dominating Set). A set of actors $\hat{D}$ is called a minimal dominating set if it satisfies Def. 2.2 and none of its proper subsets is a dominating set.

As stated above, many dominating sets can be obtained for a particular network, and in the edge case, the set of all actors also constitutes a dominating set. Another implication of the aforementioned definitions is that multiple minimum and minimal dominating sets may exist for a given multilayer network. Finally, both Def.2.2 and Def.2.3 are consistent with the original concept of the minimum dominating set introduced for classic (i.e. static, undirected, and single-layer) graphs [26], when they are applied to a single-layer network (a special case of a multilayer network with $|L| = 1$). An additional important remark is that following [27], we distinguish the **minimum** dominating set (Def.2.3) from the **minimal** dominating set (Def.2.4), as the latter can be larger than the former. This distinction is particularly important in networks for which finding D within a reasonable amount of time is infeasible.

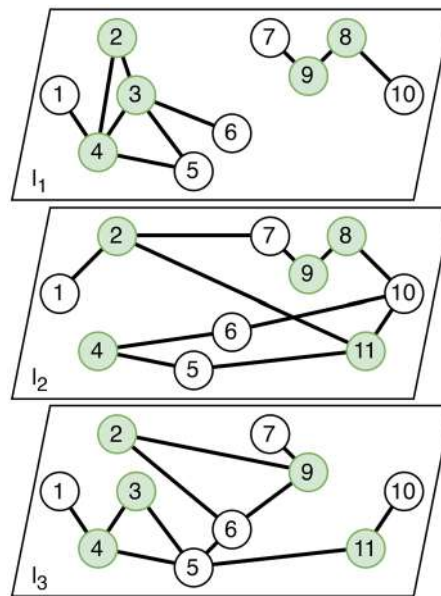


Figure 1. An example of a multilayer network with three types of relations (l_1, l_2, l_3). Nodes which represent actors that belong to the MDS are highlighted ($\hat{D} = a_2, a_3, a_4, a_8, a_9, a_{11}$).

2.3. Example

To illustrate both the MDS and an example of a multilayer network, we present Fig. 1. Note that even a small structural change can affect the MDS. For instance, adding an edge between nodes n_9 and n_6 in layer l_1 would allow actor a_3 to be removed from the MDS. The set shown in Fig. 1 satisfies both Def. 2.2 and Def. 2.4, but not Def. 2.3, as there exist dominating sets of smaller size dominating the network. Specifically, the minimum dominating sets for the given network have size 5, and eight such sets exist, e.g. $D = \{a_2, a_3, a_4, a_9, a_{10}\}$. Although the example is a simple toy case, it has been deliberately chosen to demonstrate that computing D is non-trivial. The algorithm employed in this study yields $\hat{D}$, which serves as an approximation of D in terms of size.

2.4. Related work

The problem of finding MDS was originally formulated for single-layer graphs and has since been extensively studied in the literature. The classic greedy heuristic [28] serves as a foundational method and, in addition, offers a theoretical approximation guarantee. However, in practical scenarios, it often overestimates the true MDS size due to its myopic nature, leading to suboptimal solutions. Metaheuristic approaches have been widely explored to improve upon traditional greedy approaches. The Ant Colony Optimization Algorithm hybridized with a Local Search (ACO-LS) leverages pheromone trails for guiding the search process while refining solutions through local search [29]. Building on this, an extended version, ACO-PP-LS, incorporates a pre-processing phase to generate initial solutions using a greedy heuristic, thereby accelerating convergence [30]. Beyond ACO-based methods, order-based Randomized Local Search has been proposed, representing solutions as permutations of vertices and applying a perturbation mechanism to refine dominating sets iteratively [31]. While RLS improves upon many existing methods, it remains computationally demanding and is sensitive to the choice of its perturbation operator. The Iterated Greedy (IG) algorithm has been introduced as an efficient and effective approach for finding MDS [27] to address these limitations. By iteratively destructing and reconstructing solutions while incorporating a specialized local search, IG achieves a superior balance between diversification and intensification.

Empirical evaluations demonstrate that IG not only outperforms ACO-based methods and RLS in solution quality but also significantly reduces computational effort, making it a state-of-the-art approach.

Since most existing methods focus on single-layer networks, the problem of finding the MDS in multilayer networks remains largely unexplored. One of the most recent approaches, FAST-MDSM [32], employs integer linear programming (ILP) combined with graph reduction heuristics to identify a minimal set of driver nodes across multiple interconnected layers. However, its reliance on ILP renders it computationally expensive and less adaptable to dense graphs. Notably, FAST-MDSM was designed for scale-free networks and has been primarily tested on sparse biological networks, limiting its applicability to more general settings. In this work, inspired by the IG algorithm [27], originally developed for single-layer networks, we propose a new approach for efficiently solving the MDS problem in multilayer networks.

2.5. Proposed algorithm

An extension of the local improvement routine from [27] to multilayer networks is presented as Alg. 1. As an input, it accepts a multilayer network M and an initial dominating set D' obtained using a greedy routine, as described in [33]. Note that D' is likely to be only subminimal, as it is obtained using a relatively simple heuristic; thus, the subsequent operations aim to reduce its size while ensuring that it continues to satisfy Def. 2.4.

It is important to understand the quality of the initial solution before applying the local improvement algorithm. To assess this, we refer to [34], where the authors evaluated a degree-based greedy heuristic for finding MDS in single-layer networks with $400 \leq |V| \leq 800$, for which the sizes of the minimum dominating sets were known. Experiments conducted on 60 networks revealed that the median gap between a greedily obtained MDS and the optimum is around 35%, decreasing to approximately 10% for graphs with a density of 0.5. Referring these results to the multilayer setting, which is more complex, we estimate that the initial solution provided to the local improvement routine is no closer than 35% to the optimal value.

First (l. 1–3), we compute a domination map (*ComputeDomination*) that, for each layer of M , associates nodes representing actors from D' with sets of nodes they dominate. We then initialize a flag to control the termination of the routine and assign D' as the current best solution. The next step (l. 4–27) consists of a loop that explores alternative configurations of dominating sets derived from D in an attempt to identify the smallest one. The evaluation of actors from the initial solution, as well as the exploration of potential replacements for them, is carried out in a random order with the *Shuffle* function. That is aimed to ensure that the algorithm produces different MDSs for the same graph across subsequent runs. This approach helps mitigate the risk that the results depend on a single instance of the MDS drawn for a given network.

Specifically, the *for* loop (l. 6) iterates over the actors in $\hat{D}$ to assess whether replacing a particular actor $a_{\hat{D}}$ leads to a reduction in its size. To achieve this, we identify a set of candidate replacements C for each $a_{\hat{D}}$ using the auxiliary function *FindReplacementCandidates*. For each actor $a_C \in C$, we construct a new set $\hat{D}_{\text{new}}$ by substituting $a_{\hat{D}}$ with a_C in $\hat{D}$. Then, using *IsFeasible*, $\hat{D}_{\text{new}}$ is examined to determine if it meets the criteria to dominate M . If the set remains feasible, we attempt to greedily minimize its size by iteratively removing redundant actors. If the pruned set $\hat{D}_{\text{reduced}}$ is not smaller than the current best solution, we continue iterating through C . Otherwise, we update $\hat{D}$ with $\hat{D}_{\text{reduced}}$, recompute the domination map for the updated $\hat{D}$, and reset the *improved* flag. At this point, we restart the outer loop, attempting further refinement of the dominating set. The iteration continues until no further refinements can be made. At this stage, the algorithm terminates and returns $\hat{D}$, which is locally minimal in size.

The efficiency of the proposed algorithm also plays an important role and can be estimated by assessing its complexity. First, since Alg. 1 builds on the initial solution, the time complexity of the greedy approach, $T_{\text{greedy}} = O(L |A|^2 |D'|)$, is taken as a reference. Once the local improvement

Algorithm 1 MDS with Local Improvement for Multilayer Networks

Require: $M = (A, L, V, E)$ ▷ multilayer network
Require: D' ▷ initial dominating set

```

1:  $domination \leftarrow \text{ComputeDomination}(M, D')$ 
2:  $improved \leftarrow \text{True}$ 
3:  $\hat{D} \leftarrow D'$ 
4: while  $improved$  do
5:    $improved \leftarrow \text{False}$ 
6:   for  $a_{\hat{D}} \in \text{Shuffle}(\hat{D})$  do
7:      $C \leftarrow \text{FindReplacementCandidates}(M, a_{\hat{D}}, \hat{D}, domination)$ 
8:     for  $a_C \in \text{Shuffle}(C)$  do
9:        $\hat{D}_{old} \leftarrow \hat{D}$ 
10:       $\hat{D}_{new} \leftarrow (\hat{D} \setminus \{a_{\hat{D}}\}) \cup \{a_C\}$ 
11:      if  $\text{IsFeasible}(M, \hat{D}_{new})$  then
12:         $\hat{D}_{reduced} \leftarrow \text{RemoveRedundantVertices}(M, \hat{D}_{new})$ 
13:        if  $|\hat{D}_{reduced}| < |\hat{D}_{old}|$  then
14:           $\hat{D} \leftarrow \hat{D}_{reduced}$ 
15:           $domination \leftarrow \text{ComputeDomination}(M, \hat{D})$ 
16:           $improved \leftarrow \text{True}$ 
17:          break
18:        else
19:           $\hat{D} \leftarrow \hat{D}_{old}$ 
20:        end if
21:      end if
22:    end for
23:    if  $improved$  then
24:      break
25:    end if
26:  end for
27: end while
28: return  $\hat{D}$  ▷ improved dominating set

```

phase is added, the algorithm's complexity is dictated by the search for replacements and the subsequent redundancy removal. This yields a total time complexity of $T_{\text{greedy+LI}} = O(L|A|^2|D'|^5)$ for dense graphs. Therefore, the greedy algorithm with local improvement is costlier than the naive greedy approach by a factor of $|D'|^4$. While this worst-case complexity appears computationally expensive, the algorithm's runtime in practice is often significantly lower. This practical efficiency stems from two key factors. First, the number of successful improvements found by the outer *while* loop is typically a manageable percentage of $|D'|$. Second, and more importantly, the number of candidate nodes C that can replace a node $a_{\hat{D}}$ in the dominating set is also moderate. The constraint that a candidate must dominate all nodes exclusively covered by $a_{\hat{D}}$ is highly restrictive. As a result, the inner loop rarely iterates over all $|A|$ vertices; it usually inspects just a handful of viable replacements, making the local improvement phase tractable for small- to medium-sized graphs despite the pessimistic worst-case bound.

3. EXPERIMENTAL SETUP

In this section, we describe the simulation setup, including the dataset and the diffusion model applied. We then present the baseline seed selection methods and explain how MDS was incorporated

into them. We also outline metrics used to assess diffusion effectiveness, an evaluated parameter space, and the computational resources employed.

3.1. Dataset

The efficiency of MDS depends on the structure of the underlying network. Furthermore, as mentioned in [Section 1](#), dynamics in multilayer networks often differ significantly from those observed in single-layer networks. Consequently, selecting appropriate data for experiments is crucial. To ensure a comprehensive evaluation, we utilize both real-world and artificially generated graphs. A summary of the networks used in this study, with a brief description of the domain context in which each was constructed, is provided in [Table 1](#). All artificial networks were generated using the `multinet` library [35]. The real-world networks come from different domains, but all represent human-based relationships.

Table 1. Networks used in the experiments, with their parameters and corresponding domain context

Type	Name	Layers	Actors	Nodes	Edges	Note
E-R	er-2	2	1000	2000	5459	Erdős-Rényi network [36] generated with [35].
	er-3	3	1000	3000	7136	Erdős-Rényi network [36] generated with [35].
	er-5	5	1000	5000	15 109	Erdős-Rényi network [36] generated with [35].
S-F	sf-2	2	1000	2000	4223	scale-free network [37] generated with [35].
	sf-3	3	1000	3000	5010	scale-free network [37] generated with [35].
	sf-5	5	1000	5000	10 181	scale-free network [37] generated with [35].
	arxiv	13	14 065	26 796	59 026	Coauthorship network obtained from articles published on the ‘arXiv’ repository [38].
REAL	aucs	5	61	224	620	Interactions between employees of Aarhus University, Dep. of Computer Science [39].
	ckmp	3	241	674	1370	A network depicting diffusion of innovations among physicians [40].
	lazega	3	71	212	1659	A network of various types of interactions between staff of a law corporation [41].
	l2-course	2	41	82	297	A first snapshot of the network built from interactions between students during a three-month-long abroad language course in Arabic [42].
	timik	3	61 702	102 247	881 676	A graph of interactions between users of the virtual world platform for teenagers [43].

3.2. Spreading model

We use the Multilayer Linear Threshold Model (MLTM) introduced in [44]. It simulates diffusion in a multilayer network where actors are the subject of spreading. The model uses two thresholds: μ , the traditional threshold from the classic LTM, which dictates that a node receives positive input when the sum of weighted influences from its neighbours exceeds this value, and δ (referred to as the protocol), which governs how inputs from nodes representing an actor across layers are aggregated. When the protocol threshold is exceeded, the actor (and all nodes representing it) becomes active; otherwise, the actor remains inactive, even if some of the corresponding nodes in the layers receive positive input.

At this point, it is worth clarifying the rationale behind introducing the double-threshold mechanism. Unlike studies such as [45–47], which modify baseline spreading models to more accurately simulate specific phenomena (e.g. to allow for bidirectional binary opinion changes [45]), the protocol function here serves solely as a fusion mechanism to determine the state of an actor given the states of nodes representing it. Thus, this modification should be regarded as an extension of

the classic LTM to a different network model—multilayer networks (in a manner analogous to, e.g. hypergraphs [48]), rather than as a change aimed at incorporating additional social mechanisms.

In this work, we specifically use two edge cases for the protocol function of the MLTM: *AND*, where positive input at all layers is required to activate an actor, and *OR*, where positive input at just one layer is sufficient for activation. As in [44], we will denote the set of active actors at time step t as S_t (with the seed set denoted as S_0). Moreover, to clarify the definitions of the performance metrics (see Section 3.4) employed in this study, we introduce an additional function as in Def. 3.1.

THEOREM 3.1 (Spreading Dynamics under MLTM). Let Y be a function to measure the spreading dynamics in a network M under the Multilayer Linear Threshold Model:

$$Y(M, t) = \sum_{a \in A} y_a(t),$$

where $y_a(t) \in \{0, 1\}$ denotes the state of actor a at time step t .

3.3. Seed selection methods

3.3.1. Basic heuristics

We employed five seed selection methods acting as a baseline for the further comparison with MDS: Degree Centrality (denoted as *deg-c*), Degree Discount [49] (*deg-cd*), One-Hop Neighbourhood Size [50] (*ngnb-1s*), Neighbourhood Size Discount [44] (*ngnb-sd*), and Random Choice (*random*). These methods belong to the class of rank-refining techniques, and their selection was motivated by their ability to generate a complete ranking of all actors within the network, ensuring compatibility with the application of MDS. We use each of these methods as described in [44].

3.3.2. Sorting with MDS

To evaluate the effectiveness of MDS, all of the methods above were modified to include additional steps for filtering out actors that do not belong to $\hat{D}$ (see Alg. 2). With such a setting, we were able to compare spreading efficiency on two versions of a particular seed selection method and assess the usefulness of MDS.

Selecting a seed set with MDS consists of several steps. First, the minimal dominating set $\hat{D}$ is obtained using Alg. 1. Simultaneously, a sorted list of all actors, A_ϕ , is created based on a particular heuristic ϕ listed in Section 3.3.1. The final seed set consists of the first $\lfloor s \cdot |A| \rfloor$ actors from A_ϕ that also belong to $\hat{D}$. However, particularly for large budgets, $|\hat{D}|$ can be smaller than the maximal number of actors that can be used as a seed set. In such cases, we exclude the simulation from the evaluation process since assessing the effectiveness of MDS is impossible.

It is good to note that we slightly modified Alg. 1 to address scalability issues. Specifically, for the largest evaluated network, we observed that the algorithm could not be completed within a reasonable time. To mitigate this, we introduced a timeout threshold for refining the initial solution. It was set to 5 minutes per 1000 actors, leading to a maximum runtime of over 5 hours per MDS search in *timik*.

3.4. Metrics

To assess the seed set quality, two metrics were used: *Gain* (Γ) for the overall effectiveness and the *Area under Curve* (Λ) for the dynamics of the diffusion. Their formulas are presented in Table 2. This study evaluates whether MDS improves seed sets; thus, higher Γ and Λ indicate a more effective seed set.

A way how the metrics are calculated is presented in Fig. 2. In that example, we observe that the seed set obtained with an additional MDS filtering achieves better diffusion performance in terms of the final number of activations (Γ), yet it impairs the diffusion dynamics (Λ).

Algorithm 2 Seed Selection with MDS

Require: $M = (A, L, V, E)$ ▷ multilayer network
Require: s ▷ seed set budget
Require: $\phi(M) \rightarrow A_\phi$ ▷ rank-refining seed selection heuristic
Require: $\kappa(M) \rightarrow \hat{D} : \hat{D} \subseteq A$ ▷ function returning MDS of M

```

1:  $\hat{D} \leftarrow \kappa(M)$ 
2: if  $|\hat{D}| < s$  then
3:   return NaN
4: end if
5:  $A_\phi \leftarrow \phi(M)$ 
6:  $S \leftarrow \emptyset$  ▷ seed set
7: for  $a \in A_\phi$  do
8:   if  $a \in \hat{D}$  then
9:      $S \leftarrow S \cup \{a\}$ 
10:  end if
11:  if  $|S| = s$  then
12:    break
13:  end if
14: end for
15: return  $S$ 

```

Table 2. Spreading metrics employed in the study

Symbol	Name	Description
Λ	Area under Curve	Area under a normalized curve of spreading dynamics: $\Lambda = \int_0^1 Y_{norm}(M, t) dt, \Lambda \in [0, 1]$
Γ	Gain	Total network coverage achieved by a seed set S_0 as a proportion of the actors activated during the diffusion and the total number of activatable actors: $\Gamma = \frac{ S_\infty - S_0 }{ A - S_0 }, \Gamma \in [0, 1]$.

3.5. Parameter space

To ensure the feasibility of the experiments, we selected the parameter space based on initial trials. That allowed us to exclude cases where diffusion always failed to initiate or saturated immediately. Notably, we assigned different budget sizes for the two protocols and considered node activation thresholds ranging from 0.1 to 0.9 in increments of 0.1. Note that we did not introduce heterogeneity into the model—all nodes were assigned the same threshold value. Although spreading under the MLTM is deterministic, there are many possible combinations of the MDS for each graph. To account for this, we conducted 30 independent runs for each set of evaluated parameters, drawing a new MDS in each case.

In total, each seed selection method was evaluated across 90 different MLTM parameter combinations, resulting in 10,800 scheduled experiments. Table 3 outlines the evaluated parameter space. It is worth noting that, as a follow-up to this main study, a detailed evaluation on an additional scale-free network cohort was conducted (see Section 4.4). However, to maintain consistency, we do not include the parameters of those experiments here, as they constitute a subset of those listed in Table 3.

3.6. Implementation

The experiments were implemented in Python 3.12, utilizing spreading models and seed selection methods from the `network-diffusion` library [33], version 0.17. The code was designed

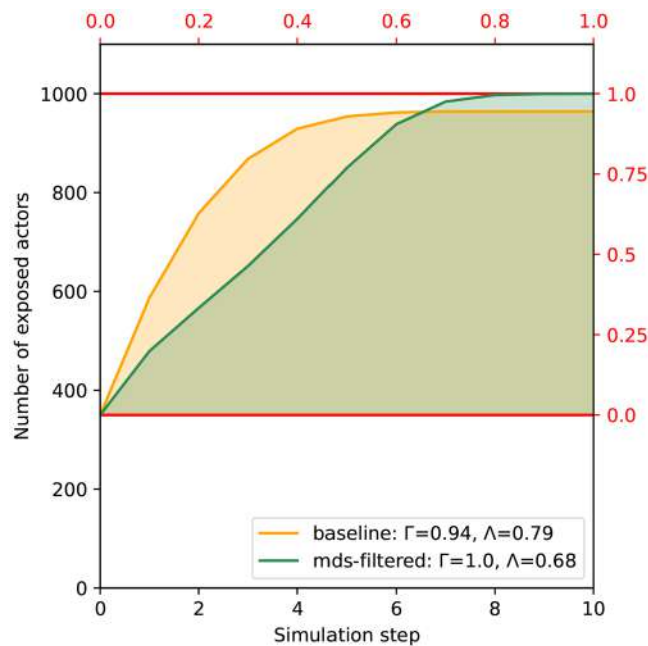


Figure 2. An example of Γ and Λ attained by the diffusion process within the *er-3* network under MLTM with $\mu = 0.4$, $\delta = AND$, initiated from a seed set selected using *neigh-sd* with $s = 35$. The black scale represents real values, whereas the red scale is used for computing Λ . Red lines indicate the number of pre-activated actors (i.e. those belonging to S_0) and the total number of actors in the network.

Table 3. Diffusion parameters evaluated in the study

Parameter	Values range	
Protocol (δ)	OR	AND
Budget (s) [% of $ A $]	{5, 10, 15, 20, 25}	{15, 20, 25, 30, 35}
Threshold (μ)	{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9}	
Seed selection method (ϕ)	{deg-c, deg-c-d, neigh-1s, neigh-sd, random}	
MDS repetitions	30	

to ensure full reproducibility, even for stochastic simulations. All experiments were conducted on a workstation running Ubuntu 20.04.4 LTS with kernel 6.5.3-arch1-1, equipped with 376 GB of memory and an Intel(R) Xeon(R) Gold 6238 CPU @ 2.10 GHz with an x86_64 architecture. The source code, data, and results are available at: <https://github.com/network-science-lab/infix-mds-ltm-mln>.

4. RESULTS ANALYSIS

We conduct the results analysis in four steps. First, we examine the MDSs obtained for each network to assess their variability with respect to the network structural properties. Next, we compare the seed sets used to initiate diffusion, focusing on the differences between those selected by the baseline heuristics and their MDS-filtered versions. Finally, based on the spreading metrics (Section 3.4), we evaluate the impact of incorporating MDS into the seed selection process. As we pay particular attention to the scale-free model, we also conduct additional experiments to ensure that the results regarding the applicability of MDS in these networks remain consistent despite variations in their configurational parameters.

Table 4. Statistical properties of MDSs and the average size reduction (*Avg. size red.*) of the initial D' by the local improvement routine obtained for each network during the experiments.^a

Network	Size range	Avg. size	Unique	IoU	Entropy	Avg. size red.
er-2	[0.26, 0.27]	0.27(3E−3)	300/300	0.42	9.13	0.15(4E−3)
er-3	[0.34, 0.36]	0.35(4E−3)	300/300	0.44	9.41	0.18(5E−3)
er-5	[0.34, 0.36]	0.35(4E−3)	300/300	0.39	9.54	0.20(5E−3)
sf-2	[0.49, 0.50]	0.50(1E−3)	300/300	0.88	9.13	0.06(2E−3)
sf-3	[0.75, 0.76]	0.76(1E−3)	298/300	0.97	9.60	0.05(1E−3)
sf-5	[0.69, 0.70]	0.70(1E−3)	300/300	0.92	9.54	0.09(1E−3)
arxiv	[1.00, 1.00]	1.00(0E−0)	1/300	NAN	NAN	0.00(0E−0)
aucs	[0.28, 0.41]	0.33(2E−2)	77/300	0.51	5.01	0.63(7E−3)
ckmp	[0.32, 0.35]	0.33(6E−3)	300/300	0.62	6.94	0.31(6E−3)
lazega	[0.14, 0.17]	0.15(9E−3)	158/300	0.31	4.71	0.25(2E−2)
l2-course	[0.20, 0.22]	0.21(1E−2)	149/300	0.34	4.36	0.13(2E−2)
timik	[0.90, 0.91]	0.91(2E−4)	295/300	0.99	15.78	0.00(3E−5)

^aFor column *Avg. size* and *Avg. size red.*, the standard deviation is reported in brackets.

4.1. Characteristics of MDSs

We begin the analysis by examining the MDSs obtained for each network during the experiments to assess their variability. The results are presented in Table 4. The first column contains the size range of MDSs obtained during the experiments, followed by the average MDS size with standard deviation, both normalized by network size. Next, we report the number of unique MDSs out of the total computed in the experiments. The fourth column provides the average intersection over union (Jaccard similarity) between all MDS pairs. Next, the entropy, calculated based on the frequency of each actor's occurrence across all MDSs, is presented. The last column shows the profit achieved by applying the local improvement routine (Alg. 1) to the dataset used in the study, compared to the greedy approach, in order to find an MDS. It is expressed as the difference between the sizes of the greedily obtained solution ($|D'|$) and the final MDS ($|\hat{D}|$), divided by the size of the former ($\frac{|D'| - |\hat{D}|}{|D'|}$), computed and averaged over the corresponding simulation runs.

It is interesting to observe the differences between Erdős-Rényi and scale-free networks. The former can be controlled with significantly fewer driver actors than the latter. In both cases, nearly every simulation returned a different MDS. However, their similarity varies: Erdős-Rényi networks exhibit a much lower Jaccard similarity compared to scale-free networks. This suggests that driver actors in that group can be assigned in a more random manner than in scale-free graphs.

With regards to real-world networks, MDS properties appear less regular. Therefore, in the following sections, we discuss the performance of the real graphs in relation to their structural similarity to both artificial types of networks. Nonetheless, the average $|\hat{D}|$ is comparable to that observed in scale-free (*timik*) and Erdős-Rényi models (*aucs*, *ckmp*, *lazega*, *l2-course*). An interesting result was obtained for *arxiv*. Due to its structural properties (on average, each actor is present in 1.91 layers, whereas the total number of layers is 13) and despite its relatively large size, the MDS for this network covers the entire set of actors. This case illustrates that, as in many aspects of network science, certain graph structures can present challenges that are difficult to address using general methods.

Entropy values appear to correlate with network size. For instance, the smallest network, *l2-course*, exhibits the lowest entropy, while the largest, *timik*, has the highest. This suggests that larger networks provide more diverse MDS configurations. Regarding the relationship between network layers and MDS size, we observe that an increase in the number of layers corresponds to a larger MDS. Finally, the number of unique MDSs is correlated with network size. For smaller networks (e.g. *aucs*, *lazega*, *l2-course*), it is likely that all possible MDS configurations were sampled.

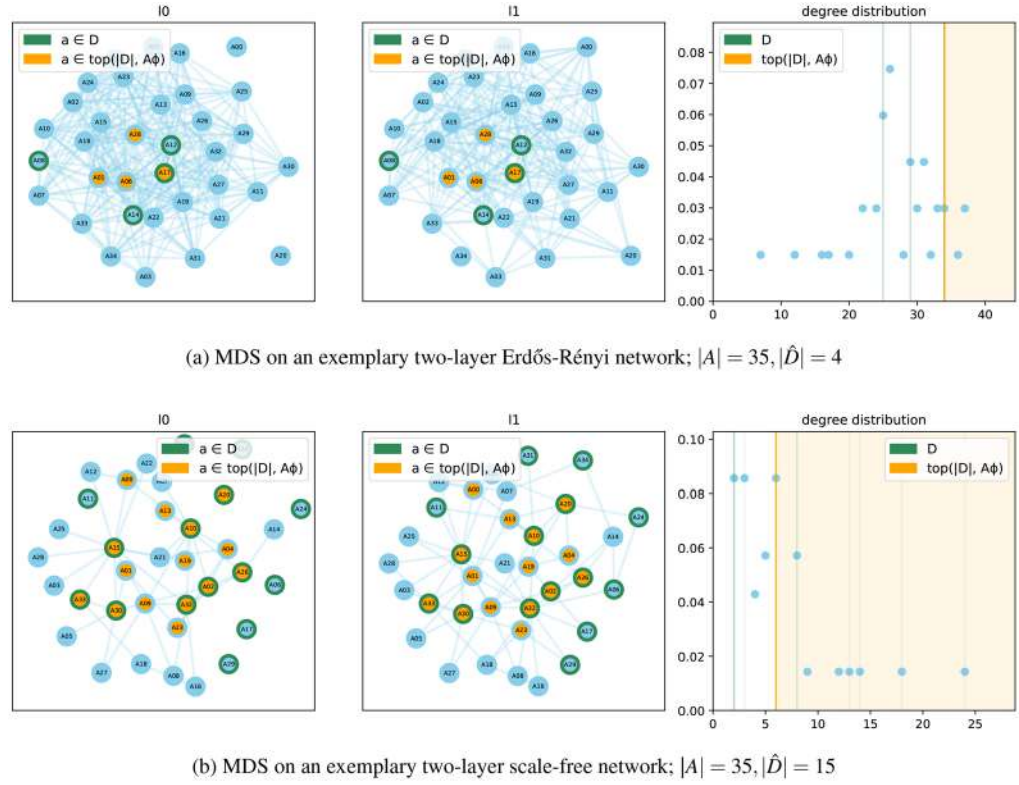


Figure 3. Typical MDS alignment with regard to the actors' degree. The first two plots depict layers of the given network, while the third shows the degree distribution. Green stripes indicate the degrees of actors belonging to the MDS; the orange area marks the degree range of a set composed of the highest-degree actors with the same size as the MDS. In (a), the MDS exhibits a much stronger correlation with the highest-degree actors compared to (b), where the MDS is larger and includes both hubs and low-degree actors.

4.1.1. Example of MDS alignment across artificial networks

Differences in MDS's properties regarding Erdős-Rényi and scale-free models can be illustrated with Fig. 3. Since the degree distribution of the latter follows a power law, controlling the network requires selecting both the hubs (e.g. actors A15, A32) and outliers, which may have connections on only some layers (e.g. A06 or A17). Consequently, a larger number of actors must be selected to exert control over the network. On the other hand, Erdős-Rényi networks, which follow a Poisson distribution, exhibit a more even degree distribution. As a result, the MDS can be smaller (e.g. $|\hat{D}| = 4$ compared to $|\hat{D}| = 15$) and selected mostly from actors with the highest degree.

4.1.2. Approximation of the globally optimal solution

Finally, to assess the robustness of Alg. 1, we analyze the extent to which it reduces the greedily obtained initial solution (as reported in the last column of Table 4). The greatest enhancement resulting from the local improvement routine was observed for the *aucs* network, where the algorithm produced an MDS that was, on average, only 37% the size of the initial solution. The only two networks for which no significant reduction in D' was achieved are *arxiv*, where MDS covers the entire network, and *timik*, due to the timeout mechanism imposed to complete the computations within a reasonable time frame. Furthermore, the scale-free nature of this network suggests that the

expected size of the MDS is inherently large. Nonetheless, one can conclude that Alg. 1 effectively reduces the size of the dominating set compared to the baseline approach.

We also sought to assess how closely the MDSs returned by Alg. 1 approximate globally optimal solutions; that is, whether the obtained minimal dominating sets are also minimum dominating sets. To this end, we executed a brute-force algorithm to exhaustively explore the solution space up to the minimal size of the MDSs found for a given network. The number of combinations to evaluate using such an approach is $\sum_{n=0}^{|\hat{D}|} \binom{|A|}{n}$.

Due to the computational complexity of this task, the evaluation was conducted on the three smallest networks—*l2-course*, *aucs*, and *lazega*. Since the number of sets to evaluate grows rapidly, we were not able to explore all possible solutions for bigger networks in a reasonable time. Nonetheless, the complete results for *l2-course* indicate that Alg. 1 was able to return the minimum dominating set. For *lazega*, we explored all combinations up to sets consisting of 8 actors; thus, recalling that $\min|\hat{D}| = 10$ was obtained with the Local Improvement routine, we conclude that $|D| \in \{9, 10\}$, making Alg. 1 a relatively efficient approach. Finally, since the number of layers in *aucs* was the largest among the three networks discussed, the best result obtained with the Local Improvement routine was $\min|\hat{D}| = 17$. We were able to explore all combinations up to 9-element sets. Therefore, $|D| \in \{10, \dots, 17\}$.

4.2. Role of MDS in seed selection

To evaluate MDS's role in seed selection, we compared all obtained seed sets (i.e. from experiments where $|\hat{D}| \geq s$). They were grouped into pairs based on the protocol, budget size, experiment repetition, and network for both the baseline and MDS-filtered methods (activation thresholds were omitted, as they do not affect seeding). The Jaccard similarity was computed for each of these pairs. Finally, the similarity scores were averaged over each network type, seed selection method, and budget size.

Table 5 presents the outcomes. A higher Jaccard similarity suggests a less prominent role of MDS, whereas lower values indicate a greater impact of MDS on the given heuristic. The results indicate that the average similarity between seed sets across all cases is approximately 0.36, demonstrating that MDS filtering consistently influences the seed selection process. However, we observe higher *IOU* values for real networks, which is due to exceptionally large MDSs in the *arxiv* and *timik* networks. Apart from these cases, similarity values are accompanied by relatively small standard deviations, which supports the observed trend. As expected, the lowest similarity between seed sets is observed for the *random* selection method, as it lacks a structured approach to choosing influential nodes. The remaining seed selection methods exhibit comparable IoU across different budget sizes, suggesting that MDS filtering affects them uniformly, thereby underscoring its universal applicability.

For both Erdős-Rényi and scale-free networks, the similarity between seed sets obtained with and without MDS filtering consistently decreases as the budget size increases (with the exception of *random*). This trend suggests that the impact of MDS filtering increases as the seeding budget grows. For smaller budget sizes, where the influence of MDS is less significant (i.e. IoU values are higher), the similarity between baseline and MDS-filtered seed sets is greater for scale-free networks than for Erdős-Rényi graphs. This observation aligns with the findings presented in Table 4, which indicate that MDS exhibits higher variability in the latter network model. Consequently, there exist more possible MDS configurations that do not always align with the baseline ranking.

In real networks, trends reflect the diverse structural patterns present in these graphs, making them less straightforward to interpret. The markedly higher standard deviation values (by an order of magnitude) further confirm the greater variability in seed set similarities within this cohort. Nonetheless, even with such unfavourable networks as *arxiv* and *timik* included, the role of MDS filtering remains both visible and significant.

Table 5. Similarities between seed sets obtained using the baseline methods and their MDS-backed variants, grouped by network type and budget size

Avg. $IOU(\phi, \phi_\kappa)$ —Erdős-Rény networks					
s	$deg-c$	$deg-cd$	$nehb-1s$	$nehb-sd$	$random$
5	0.48 (0.08)	0.50 (0.08)	0.41 (0.07)	0.42 (0.07)	0.02 (0.02)
10	0.41 (0.05)	0.43 (0.04)	0.37 (0.05)	0.35 (0.04)	0.05 (0.01)
15	0.36 (0.03)	0.39 (0.03)	0.34 (0.03)	0.33 (0.02)	0.08 (0.02)
20	0.34 (0.02)	0.36 (0.02)	0.32 (0.02)	0.32 (0.02)	0.11 (0.02)
25	0.33 (0.02)	0.34 (0.02)	0.31 (0.02)	0.31 (0.02)	0.14 (0.02)
30	0.31 (0.01)	0.32 (0.02)	0.30 (0.01)	0.29 (0.01)	0.17 (0.01)
35	0.30 (0.01)	0.31 (0.02)	0.29 (0.01)	0.29 (0.02)	0.21 (0.01)
Avg. $IOU(\phi, \phi_\kappa)$ —scale-free networks					
s	$deg-c$	$deg-cd$	$nehb-1s$	$nehb-sd$	$random$
5	0.49 (0.07)	0.53 (0.08)	0.47 (0.06)	0.46 (0.07)	0.03 (0.02)
10	0.37 (0.03)	0.38 (0.04)	0.34 (0.03)	0.35 (0.03)	0.05 (0.02)
15	0.31 (0.04)	0.34 (0.02)	0.30 (0.02)	0.32 (0.02)	0.08 (0.02)
20	0.29 (0.03)	0.32 (0.02)	0.28 (0.03)	0.30 (0.04)	0.11 (0.02)
25	0.27 (0.03)	0.30 (0.03)	0.27 (0.05)	0.29 (0.06)	0.14 (0.01)
30	0.27 (0.04)	0.29 (0.05)	0.26 (0.06)	0.30 (0.07)	0.18 (0.01)
35	0.26 (0.06)	0.28 (0.06)	0.27 (0.08)	0.30 (0.08)	0.21 (0.02)
Avg. $IOU(\phi, \phi_\kappa)$ —real networks					
s	$deg-c$	$deg-cd$	$nehb-1s$	$nehb-sd$	$random$
5	0.57(0.27)	0.61(0.23)	0.56(0.29)	0.54(0.29)	0.04(0.06)
10	0.53(0.25)	0.55(0.24)	0.49(0.28)	0.51(0.28)	0.06(0.05)
15	0.52(0.26)	0.53(0.25)	0.49(0.28)	0.50(0.28)	0.08(0.05)
20	0.53(0.27)	0.55(0.26)	0.52(0.28)	0.54(0.28)	0.11(0.04)
25	0.57(0.28)	0.57(0.27)	0.56(0.28)	0.59(0.27)	0.14(0.03)
30	0.56(0.29)	0.57(0.28)	0.55(0.29)	0.59(0.27)	0.18(0.04)
35	0.74(0.26)	0.80(0.20)	0.75(0.25)	0.82(0.18)	0.21(0.01)

4.3. Effectiveness of MDS in influence maximization

Since this study aims to evaluate the general usability of MDS for seed set selection, we have deliberately omitted the influence of specific seed selection methods. To keep the report concise, we have also aggregated the results across different network types. Figure 4, which is discussed below, presents a comprehensive summary of our experimental results in a structured format. Readers interested in a more detailed analysis are encouraged to explore the repository.

4.3.1. Feasibility of the experiments

We begin our analysis with a quantitative summary of the experiments, focusing on the second and third rows of each tile in the heatmap. The former represents the number of experiments that were executed correctly, while the latter provides insights into the cases where simulations failed. Specifically, failures occurred either due to overly restrictive diffusion parameters (left value) or when the MDS was too small to select a seed set with it (right value). It is important to note that the corresponding values of failed experiments should be identical for pairs of Γ and Λ heatmaps that represent the same dataset fold and applied protocol. Moreover, the number of cases where MDS was too small to trigger diffusion is independent of μ and δ . Consequently, these values should be the same for all experiments conducted for the given s and the network type.

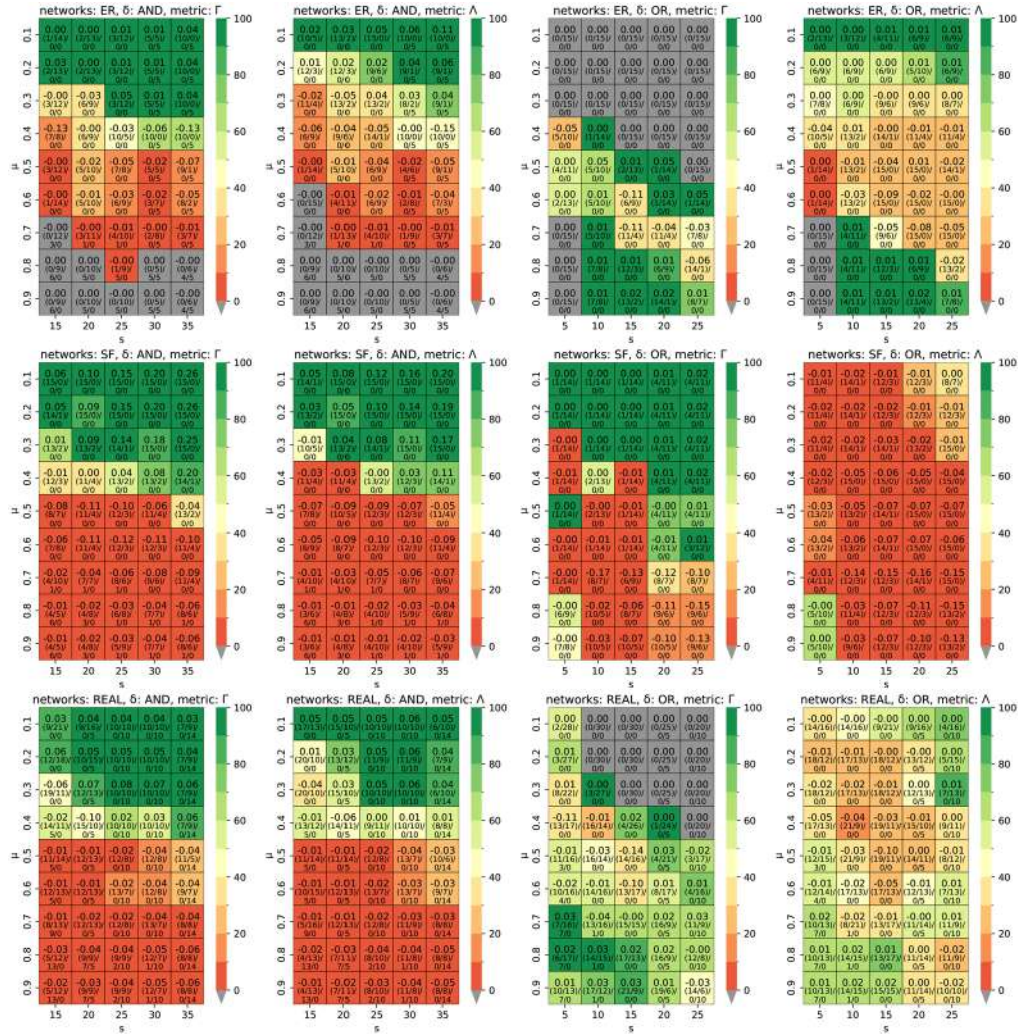


Figure 4. Improvement of MDS filtering across network types and spreading regimes, measured with Γ and Λ . Each tile shows six values: the *upper value* is the mean difference between the baseline and MDS-filtered variant ($[-1, 1]$) across feasible experiments; the *middle row* presents the count of feasible experiments, split into those with a difference greater (*left*) or less than/equal to 0.01 (*right*); the *lower row* shows unfeasible experiments where diffusion couldn't start (due to an overly strict spreading regime, *left*) or where $|\hat{D}| < s$ (*right*). The *tile colour* represents the percentage of feasible simulations where the MDS-filtered variant outperformed the baseline, computed over feasible simulations with an absolute difference greater than 0.01.

4.3.1.1. Failures due to overly demanding parameters

In Erdős-Rényi networks under the *AND* protocol, failures occur for $\mu \geq 0.7$, with their number increasing as μ grows. Conversely, as the budget increases, the number of failures decreases. In the *OR* regime, all experiments were feasible. A similar pattern is observed in scale-free networks for $\delta = \text{AND}$; however, the reduction in failed experiments due to an increasing budget occurs at a higher rate. For $\delta = \text{OR}$, all experiments were feasible. Finally, for real networks and spreading under the *AND* protocol, the region of overly strict parameters corresponds to $\mu \geq 0.4$ or a budget of $s \leq 30$. In the *OR* regime, failures occur for $\mu \geq 0.5$ or $s \leq 10$.

As expected, the number of failed experiments depends on both μ and s . The highest failure rates occur for large μ and small s , while relaxing at least one of these parameters increases the likelihood of triggering a spread. Compared to artificial networks, the subset of real networks in our study poses greater challenges in initiating the diffusion process.

4.3.1.2. Failures due to insufficient size of MDS

In Erdős-Rényi networks, for budgets $s \geq 30$, failures resulting from an insufficiently large MDS accounted for 33.33% of the cases. In contrast, in scale-free networks, the MDS was consistently large enough to select a seed set that matched the budget, and no such failures were observed. For real networks, failures of this nature began to manifest for budgets $s \geq 20$, with the frequency of these failures increasing as the budget size grew. This trend aligns with the analysis of the MDS size, as presented in Table 4.

4.3.1.3. Feasible experiments

The conditions that affect the number of feasible experiments (represented by the sum of the middle row in the tiles of the heatmaps) are the opposite of those described in the previous subsection. Specifically, a smaller threshold makes it easier to trigger the diffusion. A similar effect is observed with s , although here we encounter an obstacle in the form of the MDS size, which may be smaller than s and thus prevent the simulation from being executed.

4.3.2. Average performance differences

This part of the analysis examines the average difference (the top value on each heatmap's tile and denoted as Δ) between experiments with and without the MDS filtering operation, using both metrics employed.

4.3.2.1. Erdős-Rényi networks

Results for spreading triggered by the *AND* protocol indicate that, in general, the differences for Γ tend to increase with the budget across experiments with a similar μ . For small thresholds, the average differences in spreading effectiveness favour MDS-backed methods, while for larger μ , baseline methods tend to prevail. Similar trends are observed for $\Delta\Lambda$. For $\delta = OR$, $\Delta\Gamma$ shows no clear trends. The average difference fluctuates around zero and is generally smaller than for *AND*. For the diffusion dynamics, the trends mirror those seen for $\Delta\Gamma$.

4.3.2.2. Scale-free networks

For the *AND* protocol, significant differences are observed for Γ , with values reaching up to 0.26. As the budget increases, so does the difference, and for small thresholds, MDS-backed methods outperform the baseline. However, for larger thresholds, baseline methods prevail. For $\Delta\Lambda$, similar trends to $\Delta\Gamma$ are observed, but the values are slightly smaller. For the *OR* protocol, $\Delta\Gamma$ shows no strong trends. Positive values (favouring MDS) are very small and close to 0, while negative values exhibit a larger magnitude. For $\Delta\Lambda$, most values are negative, suggesting that using MDS even slows down the diffusion.

4.3.2.3. Real networks

For the *AND* protocol, $\Delta\Gamma$ values are smaller than those for scale-free networks but usually larger than those in observed Erdős-Rényi graphs. For small μ , MDS-backed methods tend to outperform the baseline, though less prominently than in scale-free networks, while for larger thresholds, baseline methods prevail. However, the trend of increasing differences with the budget is not so visible as in artificial graphs. A similar pattern holds for $\Delta\Lambda$. For $\delta = OR$, $\Delta\Gamma$ values are scattered around 0 with no clear trends, similar to Erdős-Rényi networks. For $\Delta\Lambda$, values are more often negative than for $\Delta\Gamma$, but no strong trends emerge as well.

4.3.3. Seed set improvement by MDS

The next part of the analysis in Fig. 4 concerns the percentage of feasible simulations where MDS-backed methods ‘significantly’ outperform their baseline versions, as indicated by colour. This comparison is intended to serve as a guideline for identifying conditions under which incorporating MDS into the seed selection process enhances diffusion, has no substantial impact, or even hinders the spreading process. In the following section, we provide a detailed discussion of each heatmap. Before that, we explain the methodology used to construct them.

For the purpose of analysis, we define a significant difference as an absolute difference according to Γ or Λ greater than 0.01 (1 percentage point) between simulations conducted on the same network and under the same spreading regime but with seed sets selected using either the baseline method (ϕ) or its MDS-backed variant (ϕ_κ). The threshold of 0.01 was chosen arbitrarily by the authors to exclude cases where the difference is negligible, i.e. when MDS neither significantly boosts nor disrupts diffusion.

To quantify the number of significantly different cases among all experiments, we split the count of feasible simulations into two values, represented in the second row of each tile: the left-hand value denotes the number of significantly different cases, while the right-hand value corresponds to cases where the difference is insignificant. The percentage of experiments in which ϕ_κ outperforms ϕ (i.e. represented by the tile colour) is computed using the former value as the denominator and, as the numerator, the count of significantly different cases where diffusion triggered with ϕ_κ was more effective than with ϕ , according to the given metric.

Finally, it is important to note that the scale is symmetrical. The greener the tile, the more frequently MDS improves the seed selection process, whereas the less green it is, the more frequently the baseline method prevails. If no significantly different cases are observed for a given combination of network type, s , and μ , the corresponding tile is displayed in grey. Naturally, MDS is not a favourable choice in such scenarios, as it provides no benefit while increasing the computational complexity.

4.3.3.1. Erdős-Rényi networks

For the AND protocol, MDS improves the overall effectiveness of diffusion in cases with small thresholds. The larger the budget, the greater the number of cases with a significant difference, and the average $\Delta\Gamma$ increases accordingly. We observe a transition boundary running diagonally across $\mu = 0.4, s = 25$, below which, regardless of s , the baseline methods remain more effective or the gain brought by MDS is insignificant. With regard to the spreading dynamics (Λ), we also observe a positive impact of MDS for diffusion with a small μ or a large s . Here, the transition between the region where baseline methods prevail is smoother, but in all cases with $\mu \geq 0.4$, classic approaches dominate.

For the OR protocol, a different trend emerges for both metrics. MDS demonstrates its superiority in two regions corresponding to the extreme values of μ . Between them, we identify a zone where baseline methods dominate, stretching diagonally from low values of both s and μ to high values of these parameters. For Γ , this trend is compressed by the region of insignificantly differing runs, and MDS exhibits stronger dominance for $\mu \geq 0.8$ and $10 \leq s \leq 20$. For Λ , the aforementioned trend is more pronounced, with the region of the most significant improvement brought by MDS resembling that observed for Γ .

4.3.3.2. Scale-free networks

Under the AND protocol, scale-free networks exhibit the greatest improvement in spreading across both metrics. As in Erdős-Rényi graphs, experiments with $\mu > 0.4$ favour baseline methods, whereas below this threshold, MDS significantly enhances diffusion, both in terms of the proportion of superior cases and the average $\Delta\Gamma$. Notably, in this region, most feasible simulations show a substantial difference in Γ between runs triggered by seed sets selected with baseline and MDS-backed methods, further reinforcing the advantage of the latter. A similar trend is observed for Λ ,

though the transition between MDS- and baseline-dominated regions appears rougher and more diagonally oriented, with $\Delta\Lambda$ generally lower than $\Delta\Gamma$.

For the OR protocol, the heatmaps resemble those of Erdős-Rényi networks, forming a diagonal zone where baseline methods dominate. This region is broader for Λ and narrower for Γ . Although the Γ heatmap contains more green-coloured tiles, the improvement brought by MDS remains limited. In many cases, the observed advantage is deceptive—the number of significantly differing results is small (at most 4 out of 15), and $\Delta\Gamma$ does not exceed 0.02. Moreover, MDS-backed methods occasionally disrupt diffusion, as seen for $\mu = 0.7, s = 10$ or $\mu = 0.8, s = 25$. The results for spreading dynamics also support this finding, providing evidence that MDS does not improve diffusion in scale-free networks under the OR protocol.

4.3.3.3. Real networks

Heatmaps for the AND protocol closely resemble those obtained for scale-free networks. However, the values of $\Delta\Gamma$ and $\Delta\Lambda$ are generally more aligned with those observed in Erdős-Rényi graphs, meaning the results are more concentrated around zero. The diagonal transition boundary between regions dominated by MDS and baseline methods appears more vertical and less abrupt.

In the OR protocol, the correlation between both heatmaps is weaker than in the AND case. Nevertheless, the same pattern seen in Erdős-Rényi networks emerges, where zones of strong MDS dominance are divided by a diagonal region stretching from small μ and s to large values of both parameters. For Γ , seed selection methods incorporating MDS filtering are favoured in most cases, but the average improvement remains modest, not exceeding 0.04. In contrast, results for Λ are more evenly distributed, with the proportion of significantly feasible cases where MDS-backed methods outperform their baseline counterparts fluctuating around 50%.

4.4. Applicability of MDS in relation to the properties of scale-free networks

As shown in Fig. 4, the usefulness of incorporating the MDS into the seed selection process depends on both the network type and the spreading conditions. The results indicate that scale-free networks benefit most from control-based approaches. Given the ubiquity of such structures in real-world systems [51], this finding is particularly relevant. However, like any synthetic model, the scale-free model has parameters that introduce additional factors. In the aforementioned experiments, we intentionally omitted these to avoid adding further variables to the evaluation. Nonetheless, since the results for scale-free networks demonstrated strong potential for MDS-backed seed selection, we conducted a follow-up experiment to assess whether this positive effect persists in these networks regardless of their configuration parameters.

4.4.1. Experiment scope

The evaluated properties of the network model had to be closely aligned with the *multinet* [35] library, which was employed in the previous parts of the study. This framework is designed to generate artificial multilayer networks and organizes configuration parameters into two groups: those governing intralayer properties, which determine the growth of graphs representing specific relations, and those defining interlayer dependencies, reflecting relations among the modelled layers. Two parameters from the configuration model were selected for evaluation, forming the basis for two cohorts of simulations. The first concerned the number of actors in the network. The second related to the number of initial nodes per layer from which the network growth process begins—these have a substantial influence on the emergence of hubs within each layer. All remaining parameters were held constant and set to typical default values.

The initial configuration setup of the network generator assumed 1,000 actors represented across three layers, each constructed using the preferential attachment model [37] with identical parameters: $m_0 = m = 6$. Regarding interlayer dependencies, the probability of adding an internal edge was set to 0.7, the probability of randomly importing an edge from another layer was 0.2, and the probability that no action occurred in a given growth step (i.e. the network grew more slowly) was

Table 6. Influence boost by incorporating MDS into the seed selection process across scale-free networks with varying numbers of actors ($|A|$), according to the overall spread (Γ) or dynamics (Λ)

	$ A = 500$	$ A = 750$	$ A = 1000$	$ A = 1250$	$ A = 1500$
$\Delta \Gamma$	0.26(0.04)	0.27(0.03)	0.26(0.03)	0.27(0.03)	0.24(0.02)
$\Delta \Lambda$	0.14(0.04)	0.13(0.03)	0.13(0.03)	0.15(0.03)	0.12(0.02)

Table 7. Influence boost by incorporating MDS into the seed selection process across scale-free networks with varying numbers of initial nodes (m_0), according to the overall spread (Γ) or dynamics (Λ)

	$m_0 = 2$	$m_0 = 4$	$m_0 = 6$	$m_0 = 8$	$m_0 = 10$
$\Delta \Gamma$	0.17(0.03)	0.23(0.03)	0.26(0.03)	0.29(0.04)	0.29(0.03)
$\Delta \Lambda$	0.14(0.03)	0.16(0.03)	0.13(0.03)	0.15(0.03)	0.11(0.03)

0.1. The dependency matrix, which served as the probability distribution for selecting the source layer from which to import an edge, was homogeneous, with values equal to $1/(|L| - 1)$, except for the diagonal entries, which were set to zero. Finally, the number of growth steps was set as $|A| - m_0$. In addition, for each utilized configuration setup, 20 instances of networks were created and used in simulations to mitigate the randomness effect.

As for the other conditions, we selected a subset of the full parameter space presented in Table 3, which appeared favourable in terms of the applicability of MDS to influence maximization in scale-free networks (see Fig. 4). All five seed selection heuristics were employed in two variants: the baseline and the MDS-backed versions. Only one spreading scenario was considered: a budget of 30% of actors, an activation threshold of 0.2, and the *AND* protocol function. Each simulated case was repeated 30 times to mitigate stochastic effects, such as those arising from obtaining the MDS.

4.4.2. Number of actors

The size of a network plays an important role in shaping its properties. To assess whether it also affects the effectiveness of using seed set drawn from the MDS, we evaluated five series of scale-free networks with varying numbers of actors: 500, 750, 1000, 1250, and 1500 (in total 1,000 networks).

Table 6 presents the experimental results, expressed as the mean difference in diffusion performance, measured in terms of overall spread (Γ) and dynamics (Λ), between corresponding runs using MDS-backed and baseline seed selection heuristics. As can be observed, both metrics are similar across the network cohorts, indicating that the applicability of MDS does not depend on the number of actors in the considered spreading scenario. Moreover, the improvement in spreading is consistent and in line with the results depicted in Fig. 4.

4.4.3. Number of initial nodes

The second parameter of the scale-free model we evaluated was m_0 . It governs the number of initial nodes used in the first step of the graph generation process and, consequently, is closely related to the expected number of hubs. As we consider multilayer networks, m_0 could take different values in each layer. However, to facilitate more straightforward conclusions, we chose to keep it uniform across all layers. The number of links created by a newly added node (m) was also fixed, with $m = m_0$.

The results are presented in Table 7, following the format used in Section 4.4.2. One can note that they are broadly consistent with the trends observed in Fig. 4, although interpretation is less straightforward due to one anomaly. From the perspective of Γ , the benefit of incorporating the MDS, while consistently substantial, diminishes as m_0 decreases. This may be because, for low values of this parameter, hubs are more pronounced, and the baseline seed selection heuristics, which largely rely on centrality measures, can already identify seed nodes effectively. As the number of hubs increases, however, the role of the MDS becomes more prominent, as it helps focus on the most influential hubs.

5. CONCLUSIONS

In this work, we examined the impact of using MDS for seed selection in a problem of influence maximization in multilayer networks under the Linear Threshold Model. We conducted a comprehensive analysis across five rank-refining seed selection heuristics, eleven networks, and a wide range of diffusion regimes. Additionally, we adapted the local improvement algorithm for finding MDS to the multilayer network setting.

5.1. Key findings

After analyzing the results, we conclude that using MDS in seed selection process benefits diffusion under low activation thresholds (up to $\mu = 0.4$) and the $\delta = \text{AND}$, regardless of the network type. In real-world scenarios, this spreading regime may correspond to situations where convincing someone requires fewer acquaintances, but these connections must span all layers (see the election example in [Section 1](#)). Higher seeding budgets further amplify this effect, as supported by [Table 5](#), which shows that the role of MDS becomes more significant with increasing s . The underlying reason is that MDS distributes seeds more widely across the network, including its periphery, whereas baseline methods concentrate them in the core. At lower μ , this ensures faster coverage of the entire network. Baseline methods, confined to the core, lack seeds in the periphery. As the threshold increases, MDS-backed methods cease to be effective sooner, as their influence is split between the core and periphery.

Under the OR protocol, where spreading is inherently easier, the effects observed in AND are far less pronounced. A seed set strong on a single layer is often sufficient to drive diffusion across the entire network, which baseline methods achieve more effectively. However, when traditional approaches fail under high thresholds, MDS can offer an advantage. In situations where finding seed set through standard methods becomes challenging, MDS facilitates diffusion that would otherwise not occur.

We also observe that scale-free networks are the most susceptible to diffusion improvement with MDS. This stems from the presence of both hub and peripheral nodes in MDS ([Fig. 3](#)). Unlike Erdős-Rényi networks, which distribute connections more evenly, scale-free structures benefit from seed sets that span both the core and periphery, making MDS-backed methods more effective in triggering diffusion. Moreover, the role of MDS in seed selection is the most pronounced for this networks compared to the others ([Table 5](#)) and persists regardless of its parameters ([Tables 6 and 7](#)). In contrast, Erdős-Rényi networks are less suitable for MDS-based approaches due to their degree distribution, which results in MDS being more randomized and typically too small (see [Table 4](#)) to meet the budget requirements.

For the real networks analyzed, MDS sizes ([Table 4](#)) suggest this subset includes both networks resembling scale-free and Erdős-Rényi models, making it the most representative of real-world scenarios. As shown in [Fig. 4](#), MDS offers the greatest benefit under the AND protocol with lower activation thresholds, improving overall coverage and diffusion dynamics. However, the example of *arxiv* shows that, in some cases, using the MDS is meaningless when it covers the entire set of actors.

Finally, despite effectiveness, using MDS comes with additional computational costs ([Section 3.3.2](#)) that need to be taken into consideration when choosing a seed selection approach. Nonetheless, in some cases, the increase in Λ or Γ might be big enough to justify the additional time needed to compute MDS.

5.2. Limitations of the study and next steps

This study includes a diverse set of networks, comprising both real-world examples and representatives of scale-free and Erdős-Rényi graphs. However, a more in-depth evaluation of the artificial models employed may provide additional insight, particularly in the case of the latter model, which was omitted owing to discouraging results. Future work may also explore the applicability of our findings to alternative graph structures, such as small-world networks. Similarly, the impact

of adopting different spreading models, such as the Independent Cascade, could be investigated. Addressing these limitations thus presents a promising direction for further research.

Another possible avenue for future work is to explore alternative interpretations of the control problem in multilayer networks. Since this graph model is built upon the idea of capturing multiple types of relations among a set of agents, Def. 2.2 was formulated such that an actor must be dominated in all (as opposed to just any) layers of the network in order to be considered controlled. Nevertheless, one can conceive an alternative formulation in which an actor is regarded as dominated if it is linked to a dominating actor in at least one layer. The MDS derived under this relaxed condition would typically be smaller than those obtained using the stricter formulation adopted in this work, and the resulting influence dynamics may also differ.

While the steps outlined here are only examples of potential directions for deeper exploration of the problem undertaken in this work, the primary conclusion remains clear: if one seeks to persuade a wide audience, particularly those whose conviction is easily swayed but must be thoroughly convinced across all aspects of their lives, starting from MDS is a good choice.

CONFLICT OF INTEREST

None declared.

FUNDING

This work was supported by the Australian Research Council [grant no. DP190101087]; the National Science Centre, Poland [grant no. 2022/45/B/ST6/04145]; the Polish Ministry of Science and Higher Education programme 'International Projects Co-Funded'; and the EU under the Horizon Europe [grant no. 101086321]. Views and opinions expressed are those of the authors and do not necessarily reflect those of the funding agencies.

REFERENCES

1. Dickison ME, Magnani M, Rossi L. *Multilayer Social Networks*. Cambridge, England, UK: Cambridge University Press, 2016.
2. Kivelä M, Arenas A, Barthélemy M *et al*. Multilayer networks. *J Complex Netw* 2014;**2**:203–71.
3. Salehi M, Sharma R, Marzolla M *et al*. Spreading processes in multilayer networks. *IEEE Trans Netw Sci Eng* 2015;**2**:65–83.
4. Granovetter M. Threshold models of collective behavior. *Am J Sociol* 1978;**83**:1420–43.
5. Watts DJ. A simple model of global cascades on random networks. *Proc Natl Acad Sci USA* 2002;**99**:5766–71.
6. Goldenberg J, Libai B, Muller E. Talk of the network: a complex systems look at the underlying process of word-of-mouth. *Mark Lett* 2001;**12**:211–23.
7. Erlandsson F, Bródka P, Borg A. Seed selection for information cascade in multilayer networks. In: *Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications)*. Springer, 2018, 426–36.
8. Bródka P, Jankowski J, Michalski R. Sequential seeding in multilayer networks. *Chaos* 2021;**31**:033130.
9. Lin W, Xu Q, Li Y *et al*. A tensor-based independent cascade model for finding influential links considering the similarity. *Chaos Solitons Fractals* 2023;**173**:113655.
10. Achour O, Romdhane LB. A theoretical review on multiplex influence maximization models: theories, methods, challenges, and future directions. *Expert Syst Appl* 2025;**266**:125990.
11. Yağan O, Gligor V. Analysis of complex contagions in random multiplex networks. *Phys Rev E Stat Nonlin Soft Matter Phys* 2012;**86**:036103.
12. Zhong YD, Srivastava V, Leonard NE. Influence spread in the heterogeneous multiplex linear threshold model. *IEEE Trans Control Netw Syst* 2022;**9**:1080–91.
13. Kempe D, Kleinberg J, Tardos É. Maximizing the spread of influence through a social network. In: *9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, 137–46.
14. Singh SS, Srivastava D, Verma M *et al*. Influence maximization frameworks, performance, challenges and directions on social network: a theoretical study. *J King Saud Univ-Comput Inf Sci* 2022;**34**:7570–603.
15. K VR, Chowdary CR. K++ Shell: influence maximization in multilayer networks using community detection. *Comput Netw* 2023;**234**:109916.

16. Chen Y, Wang W, Feng J et al. Maximizing multiple influences and fair seed allocation on multilayer social networks. *PLoS One* 2020;**15**:e0229201.
17. Yuan Z, Shao M, Chen Z. Graph Bayesian optimization for multiplex influence maximization. *AAAI* 2024;**38**:22475–83.
18. Shu J, Zou Y, Cui H et al. Node importance evaluation in heterogeneous network based on attention mechanism and graph contrastive learning. *Neurocomputing* 2025;**626**:129555.
19. Liu Y-Y, Slotine J-J, Barabási A-L. Controllability of complex networks. *Nature* 2011;**473**:167–73.
20. Liu Y-Y, Barabási A-L. Control principles of complex systems. *Rev Mod Phys* 2016;**88**:035006.
21. Doostmohammadian M. Minimal driver nodes for structural controllability of large-scale dynamical systems: node classification. *IEEE Syst J* 2020;**14**:4209–16.
22. Jiang L, Tang L, Lü J. Controllability of multilayer networks. *Asian J Control* 2022;**24**:1517–27.
23. Sadaf A, Mathieson L, Bródka P, Musial K. Maximising influence spread in complex networks by utilising community-based driver nodes as seeds. In: *Annual International Conference on Information Management and Big Data*. Springer, 2022, 126–41.
24. Sadaf A, Mathieson L, Bródka P et al. A bridge between influence models and control methods. *Appl Netw Sci* 2024;**9**:38–1.
25. Karp RM. *Reducibility among Combinatorial Problems*. Springer US, 1972. 85–103.
26. Nacher JC, Akutsu T. Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control. *New J Phys* 2012;**14**:073005.
27. Casado A, Bermudo S, López-Sánchez A et al. An iterated greedy algorithm for finding the minimum dominating set in graphs. *Math Comput Simul* 2023;**207**:41–58.
28. Chvatal V. A greedy heuristic for the set-covering problem. *Math Operat Res* 1979;**4**:233–5.
29. Potluri A, Singh A. Two hybrid meta-heuristic approaches for minimum dominating set problem. In: *Swarm, Evolutionary, and Memetic Computing*. Berlin, Heidelberg: Springer, 2011, 97–104.
30. Potluri A, Singh A. Hybrid metaheuristic algorithms for minimum weight dominating set. *Appl Soft Comput* 2013;**13**:76–88.
31. Chalupa D. An order-based algorithm for minimum dominating set with application in graph mining. *Inf Sci* 2018;**426**:101–16.
32. Nacher JC, Ishitsuka M, Miyazaki S et al. Finding and analysing the minimum set of driver nodes required to control multilayer networks. *Sci Rep* 2019;**9**:576–12.
33. Czuba M, Nurek M, Serwata D et al. Network diffusion framework to simulate spreading processes in complex networks. *Big Data Min Anal* 2024;**7**:637–54.
34. Sanchis LA. Experimental analysis of heuristic algorithms for the dominating set problem. *Algorithmica* 2002;**33**:3–18.
35. Magnani M, Rossi L, Vega D. Analysis of multiplex social networks with R. *J Stat Soft* 2021;**98**:1–30.
36. Erdős P, Rényi A et al. On the evolution of random graphs. *Publ Math Inst Hung Acad Sci* 1960;**5**:17–60.
37. Bollobas B, Borgs C, Chayes J, Riordan O. Directed scale-free graphs. In: *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003, 132–9.
38. De Domenico M, Lancichinetti A, Arenas A et al. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys Rev X* 2015;**5**:011027.
39. Rossi L, Magnani M. Towards effective visual analytics on multiplex and multilayer networks. *Chaos Solitons Fractals* 2015;**72**:68–76.
40. Coleman J, Katz E, Menzel H. The diffusion of an innovation among physicians. *Sociometry* 1957;**20**:253–70.
41. Snijders TAB, Pattison PE, Robins GL et al. New specifications for exponential random graph models. *Sociol Methodol* 2006;**36**:99–153.
42. Paradowski MB, Whitby N, Czuba M et al. Peer interaction dynamics and second language learning trajectories during study abroad: a longitudinal investigation using dynamic computational social network analysis. *Lang Learn* 2024;**74**:58–115.
43. Jankowski J, Michalski R, Bródka P. A multilayer network dataset of interaction and influence spreading in a virtual world. *Sci Data* 2017;**4**:170144.
44. Czuba M, Bródka P. Rank-refining seed selection methods for budget constrained influence maximisation in multilayer networks under linear threshold model. *Soc Netw Anal Min* 2025;**15**:46.
45. Ke X, Meng F. Opinion dynamics under double thresholds. *Phys Lett A* 2024;**513**:129601.
46. Nowak B, Sznajd-Weron K. Symmetrical threshold model with independence on random graphs. *Phys Rev E* 2020;**101**:052316.
47. Abella D, González-Avella JC, San Miguel M et al. Ordering dynamics and aging in the symmetrical threshold model. *New J Phys* 2024;**26**:013033.
48. Chen L, Zhu Y, Zhu J et al. A simple model of global cascades on random hypergraphs. *Chaos Solitons Fractals* 2025;**193**:116081.

49. Chen W, Wang Y, Yang S. Efficient influence maximization in social networks. In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09*, New York, NY, USA: Association for Computing Machinery, 2009, 199–208.
50. Magnani M, Rossi L. The ML-model for multi-layer social networks. In: *2011 International Conference on Advances in Social Networks Analysis and Mining*, 2011, 5–12.
51. Barabási A-L. Scale-free networks: a decade and beyond. *Science* 2009;**325**:412–3.

4.6 Identifying Super Spreaders in Multilayer Networks

Authors: [Michał Czuba](#), Mateusz Stolarski, Adam Piróg, Piotr Bielak, Piotr Bródka

Published at: 28th European Conference of Artificial Intelligence; October 2025

MNiSW points: 140 (Core A)

Citations: Web of Science: 0

(as of 6th March 2026) Scopus: 0

Google Scholar: 1

CRedit:

Conceptualisation Methodology Software Validation
Formal analysis Investigation Data curation
Writing – original draft Writing – review & editing
Visualisation Project administration

Identifying Super Spreaders in Multilayer Networks

Michał Czuba^{a,1,*}, Mateusz Stolarski^{a,1}, Adam Piróg^b, Piotr Bielak^a and Piotr Bródka^a

^aWrocław University of Science and Technology, Wrocław, Poland

^b4Semantics, Warsaw, Poland

Abstract. Identifying super-spreaders can be framed as a subtask of the influence maximisation problem. It seeks to pinpoint agents within a network that, if selected as single diffusion seeds, disseminate information most effectively. Multilayer networks, a specific class of heterogeneous graphs, can capture diverse types of interactions (e.g., physical-virtual or professional-social), and thus offer a more accurate representation of complex relational structures. In this work, we introduce a novel approach to identifying super-spreaders in such networks by leveraging graph neural networks. To this end, we construct a dataset by simulating information diffusion across hundreds of networks — to the best of our knowledge, the first of its kind tailored specifically to multilayer networks. We further formulate the task as a variation of the ranking prediction problem based on a four-dimensional vector that quantifies each agent's spreading potential: (i) the number of activations; (ii) the duration of the diffusion process; (iii) the peak number of activations; and (iv) the simulation step at which this peak occurs. Our model, *Top-SpreadersNetwork*, comprises a relationship-agnostic encoder and a custom aggregation layer. This design enables generalisation to previously unseen data and adapts to varying graph sizes. In an extensive evaluation, we compare our model against classic centrality-based heuristics and competitive deep learning methods. The results, obtained across a broad spectrum of real-world and synthetic multilayer networks, demonstrate that *TopSpreadersNetwork* achieves superior performance in identifying high-impact nodes, while also offering improved interpretability through its structured output.

1 Introduction

Spreading phenomena in networks are investigated across various domains, including viral marketing, transportation systems, epidemiology, or social media [38]. These studies typically assume the propagation of a certain phenomenon over a graph-like structure simulating a real-world system. A key challenge in this context is to identify the agents that have the greatest impact on the spread. The significance of this task is underscored by its diverse applications, such as detecting vulnerable nodes in power networks, scheduling targeted immunisations, or selecting influencers for advertising campaigns. Nonetheless, addressing such problems requires accurate models of the systems under consideration. To this end, multilayer networks have been proposed [24]. These models provide a more nuanced representation of relationships by enabling, for instance, social ties to be split across multiple layers that correspond to different modes of

interaction (e.g., physical, virtual). Considering spreading phenomena within a multilayer setting can better capture the complexity of real-world systems, leading to more precise results.

1.1 Related works

We review prior work on techniques to maximise diffusion in complex (particularly multilayer) networks, encompassing both classic and learning-based methods. As a prelude, we first discuss graph neural networks and their capacity to model network dynamics.

Graph neural networks. Graph Neural Networks (GNNs) [47] represent a category of deep learning techniques specifically developed for inference on graph-structured data. The standard architecture of GNNs involves an iterative process of exchanging transformed node features and aggregating information from neighbouring nodes (so called message-passing). In a GNN with k layers, each node incorporates information from its k -hop neighbourhood. Variations in the design of the aggregation function and the combination function give rise to different GNN architectures [23, 7]. The resulting high-level node or graph representations are employed across a range of tasks. GNNs have been successfully applied to problems such as estimating information diffusion [5], graph-based reasoning [27], identifying the source of diffusion on graphs [45, 28], or even deep graph generation [29]. By leveraging proper information fusion [2], GNNs present a powerful solution to model graph-based data even on multilayer networks.

Influence maximisation problem. The problem of influence maximisation was originally introduced as a combinatorial optimisation task by [22], laying the foundation for a substantial body of research and practical applications over the following decades. It aims to identify a set of agents which, when used as diffusion seeds, maximise the spread of a given process across the network. The majority of traditional (i.e., non-learning-based) approaches can be classified into three categories: simulation-based, proxy-based, and heuristic-based. These methods often rely on centrality-based heuristics, utilising them directly, e.g., selecting top- k nodes with the highest degree centrality, or building upon them, e.g., with discounting algorithms [6]. Although computationally efficient and interpretable, they operate under local assumptions that limit their adaptability to complex diffusion dynamics. For a comprehensive overview of these techniques, we refer the reader to a survey [40].

In recent years, increasing attention has been devoted to addressing influence maximisation problem with learning-based approaches. These methods employ deep learning techniques to overcome the limitations of conventional methods, particularly their poor generalisation capabilities. Within this paradigm, several approaches can

* Corresponding Author. Email: michal.czuba@pwr.edu.pl

¹ Equal contribution.

be found which model the nodes influence, such as leveraging the latent nodes embeddings [42, 25], or even incorporate reinforcement learning into influence maximisation frameworks [43, 30]. For instance, DeepIM [30] addresses this task through a transductive, generative encoder-decoder architecture augmented with knowledge distillation. While effective, this approach relies on full access to the graph structure during training, which limits its ability.

Nevertheless, most research on influence maximisation continues to focus on the information diffusion within homogeneous network structures, which offer limited expressiveness. More recently, however, attention has begun to shift towards more complex network models, opening new research directions. For instance, Heterogeneous Influence Maximisation incorporates diverse node and edge types by leveraging meta-path-based methods [12, 26], while Multilayer Influence Maximisation explores information diffusion across multiple platforms via shared nodes [21].

As in the case of homogeneous graphs, many approaches in multilayer networks rely on heuristics grounded in a notion of centrality, often adapted from single-layer settings. For instance, neighbourhood-size [33] and its discounting variant [9] are heuristics related to degree centrality. More recently, learning-based methods have also been developed to address the problem in such networks. For instance, GBIM [50] formulates multiplex influence maximisation as a graph Bayesian optimisation problem, employing an item-association network to govern the diffusion process. While conceptually appealing, the approach suffers from scalability issues in practice. Models like GNNRank [17] and GraphTR [31], while potentially suitable for ranking tasks in heterogeneous graphs, suffer from similar transductive constraints. Unsupervised node embedding methods are also potentially applicable in this context, most notably Multi-node2vec [46], which learns continuous actor representations in multilayer networks. These embeddings can subsequently be clustered to estimate influence scores based on structural proximity. However, such two-stage pipelines introduce additional hyperparameter sensitivity and may lack end-to-end optimisation.

Conclusions. A review of the literature reveals that multilayer networks remain underexplored compared to classic graphs in the context of spreading processes, especially influence maximisation techniques, particularly those grounded in machine learning paradigms. Taken together, these limitations highlight a pressing need for robust, generalisable, and inductive methods addressing these challenges.

1.2 Assumptions and main contributions

In this study, we employ GNNs to model latent diffusion dynamics in multilayer networks and develop an end-to-end framework for identifying super-spreaders. This task can be viewed as a variant of the influence maximisation problem, which seeks to identify those agents within a graph that, if used as a single element seed set, are able to disseminate information in the most effective manner — either by reaching the largest portion of the network or delivering it in the shortest possible time. Super-spreaders play a role analogous to that of a playmaker in a football team: if such a player is removed or underperforms, the team's structure becomes destabilised and its winning potential is immediately diminished.

We formulate the problem inspired by learning-to-rank task, in which the model ranks actors according to their spreading potential score, based on four features: (i) the number of activations; (ii) the duration of the diffusion process; (iii) the peak number of activations; and (iv) the simulation step at which this peak occurs — predicted for a setting when the actor serves as the sole seed of the diffusion.

Since, to the best of our knowledge, no such dataset exists, we construct one by generating hundreds of synthetic networks and simulating diffusion under the Independent Cascade Model [15] adapted to multilayer networks, to capture the aforementioned vectors.

Next, we propose a novel approach for identifying super-spreaders in multilayer networks by introducing the *TopSpreadersNetwork* (*ts-net*) — a robust GNN capable of generalising to previously unseen data, a feature rarely found among competitive trainable solutions. This ability stems from an architectural design that enables training on a diverse collection of networks, allowing the model to capture transferable structural patterns. We demonstrate the practical applicability and scalability of the proposed solution by predicting top-spreaders in real-world networks of varying sizes, with no constraints on the number of actors or layers. Furthermore, a comparison with both classical heuristics and state-of-the-art trainable models shows that *ts-net* consistently outperforms competitive solutions. The source code is available at: <https://github.com/network-science-lab/infmax-trainer-icm-mln>, appendices are published as [10].

In summary, this work contributes by introducing:

- an inductive GNN model for super-spreader identification in multilayer networks, *TopSpreadersNetwork*, capable of processing networks with varying sizes of actors and relations;
- the *TopSpreadersDataset* constructed from simulations on synthetic and real-world multilayer networks, capturing the spreading potentials of actors;
- alternative approaches for data transformations and embedding fusion that enhance robustness in the ranking prediction task from multirelational data.

2 Problem definition

Before presenting the proposed model and dataset, we first need to introduce a formal definition of the problem under consideration, as well as the data structure relevant to this study.

2.1 Multilayer networks

To express our understanding of multilayer networks, we adopt the notation introduced by [24], as shown in Def. 1.

Definition 1 (Multilayer network). *A multilayer network can be described as a tuple $M = (A, L, V, E)$ comprising the following sets:*

- actors $A = \{a_1, a_2, \dots, a_{|A|}\}$,
- layers $L = \{l_1, l_2, \dots, l_{|L|}\}$,
- nodes $V = \{v_1^1, v_1^2, \dots, v_{|A|}^1, v_{|A|}^2, \dots\} : V \subseteq A \times L$,
- edges $E = \{(v_1^1, v_2^1), \dots, (v_1^2, v_2^2), \dots\} : (v_1^1, v_2^2) \notin E \wedge (v_1^1, v_2^1) \equiv (v_2^1, v_1^1)$.

In essence, a multilayer network comprises a collection of intra-connected single-layer networks, where each actor may be represented by at most one node per layer, i.e., $V \subseteq A \times L$. For example, the representation of actor a_1 in layer l_2 is given by the node v_1^2 . The model disallows interlayer edges, meaning that connections can occur only between nodes within the same layer: $(v_1^1, v_2^2) \notin E$. Furthermore, to simplify the analysis, all edges are assumed to be undirected and unweighted, that is, $(v_1^1, v_2^1) \equiv (v_2^1, v_1^1)$. At this point, it is worth noting that the concept of multilayer networks partially overlaps with that of heterogeneous graphs [39], the latter allowing a richer representation of the actor set (e.g., distinguishing between different classes of nodes) and permitting interlayer connections.

2.2 Top-spreaders identification

Since the goal of this study is to find an effective and applicable solution to the problem of super-spreaders identification, we must first formulate our approach to this task. To begin with, let us define a model ϕ that produces a ranking of actors (Def. 2) and a way such a ranking is created to reflect their spreading power (Def. 3).

Definition 2 (Ranking predictor). *Let $\phi(M) \rightarrow \hat{\mathbf{R}}$ be a function that predicts a ranking $\hat{\mathbf{R}}$ of actors from a multilayer network M .*

Definition 3 (Top-spreaders ranking). *A top-spreaders ranking R is built upon a function $\text{score} : a \rightarrow \mathbb{R}^+$ that maps an actor a to a value reflecting its spreading potential, defined as:*

$$\mathbf{R} = [a_i, a_j, \dots, a_k] : \text{score}(a_i) \geq \text{score}(a_j) \geq \dots \geq \text{score}(a_k)$$

It is important to note that, by introducing ground-truth ranking $\mathbf{R}$ and basing subsequent parts of the work on this notion (especially the methods used to construct ground-truth rankings), we deviate from the classic influence maximisation problem. This departure is justified by the non-additivity of seed sets: the top- k actors or nodes that are most effective at spreading information individually do not necessarily constitute the optimal k -element seed set [22].

Another important remark concerns the comparison between the ground-truth $\mathbf{R}$ and the predicted ranking $\hat{\mathbf{R}}$. A natural approach is to employ standard ranking-quality metrics such as Average Precision@ k , or even the Jaccard index@ k . However, in the problem we aim to address, the primary focus lies on the accuracy of the topmost predictions, where the most influential spreaders are expected to appear. Consequently, such metrics may be overly strict in evaluating discrepancies at the top of the ranking. For example, if the predicted ranking places the third and fourth most influential actors in the top two positions, it will yield the same Jaccard index as predicting two completely irrelevant actors in those positions, despite the former being a much more acceptable error.

To address this limitation and ensure a fairer comparison, we introduce the notion of a cumulated spreading score for the top- k spreaders, as formalised in Def. 4:

Definition 4 (Cumulated spreading score). *The cumulated score of the top- k spreaders in a network M is defined as:*

$$y(\mathbf{R}, k) = \sum_{a \in \mathbf{R}[:k]} \text{score}(a)$$

where:

$\mathbf{R}[:k]$ - first k elements of the top-spreaders ranking list.

Based on the above, we can assess the quality of a predicted ranking $\hat{\mathbf{R}}$ in relation to the ground truth $\mathbf{R}$ (Def. 5). Instead of evaluating the prediction solely by the ordering of $\hat{\mathbf{R}}$, we compare the spreading scores of the top- k actors in $\hat{\mathbf{R}}$ with those of the top- k actors in $\mathbf{R}$.

Definition 5 (Relative cumulated spreading score). *The relative cumulated score of a predicted top- k actors from a ranking $\hat{\mathbf{R}}$ w.r.t. the ground-truth ranking $\mathbf{R}$ is given by:*

$$y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, k) = \frac{y(\hat{\mathbf{R}}, k)}{y(\mathbf{R}, k)}, \quad y_{rel}(\cdot) \in [0, 1]$$

Finally, we summarise the objective of this study in formal terms by introducing the optimisation task presented in Def. 6, which aims to minimise the difference between $\mathbf{R}$ and $\hat{\mathbf{R}}$.

Definition 6 (Top-spreaders identification problem). *The task of identifying the top- k spreaders in a network M , given a ground-truth ranking $\mathbf{R}$, can be formulated as finding a function ϕ satisfying the following criterion:*

$$\phi(M) \rightarrow \hat{\mathbf{R}} : \min |1 - y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, k)|$$

3 Data preparation

To facilitate the task of identifying super-spreaders in multilayer graphs, we first constructed a ground-truth dataset. The process consisted of three stages: collecting the graphs, simulating diffusion, and deriving a spreading potential score from the recorded sub-metrics.

3.1 Collecting the networks

We began by compiling a set of real-world multilayer graphs that are widely used within the network science community. In addition, using the `multinet` library [34], we generated one hundred Erdős-Rényi [13] and one hundred Preferential Attachment [3] networks, supplementing these with eight graphs previously used in our own research. Tab. 1 presents collected networks with their basic parameters shortlisted.

Table 1: Networks comprising the *TopSpreadersDataset*, with their key parameters: number of layers, actors, nodes, edges, and the average degree per actor. For artificial graphs, mean values are reported.

Network type	Layers	Actors	Nodes	Edges	Degree
artificial-er	3.52	558.19	1741.70	6684.00	24.13
artificial-pa	3.52	574.51	1976.07	42636.53	122.10
artificial-small	2.75	1000.00	2750.00	6609.12	13.22
arxiv [11]	13	14065	26796	59026	8.39
aucs [37]	5	61	224	620	20.33
ckmp [8]	3	241	674	1370	11.37
eu-trans [4]	37	417	2034	3588	17.21
l2-course [36]	2	41	82	297	14.49
lazega [41]	3	71	212	1659	46.73
timik [20]	3	61702	102247	875191	28.37

3.2 Simulating information spreading

The second step involved simulating diffusion under the Independent Cascade Model (ICM) [15], adapted to account for the multilayer structure (MICM). Each actor in the network was used as a single-element seed set. During each simulation, we recorded four metrics that capture the spreading potential of the initiating actor, following [42]. These form a spreading-potential vector (Def. 7).

Definition 7 (Spreading-potential vector). *A vector of spreading potentials can be defined for each actor from the network M as follows:*

$$\mathbf{p} = [p_{ex}, p_{sl}, p_{pi}, p_{pl}]$$

where:

- p_{ex} - total number of activated actors across the network,
- p_{sl} - spreading duration, as a number of MICM iterations,
- p_{pi} - maximum number of actors activated simultaneously,
- p_{pl} - iteration at which the activation peak occurred.

Enhancement of the ICM to multilayer graphs has been carried out based on the same premise as the extension of its similarly popular counterpart — Linear Threshold Model (LTM), proposed by [51].

Following the authors, we assume that actors are the primary subjects of the diffusion process, while the nodes representing them across different layers play an auxiliary role. In the classic ICM, a single parameter π controls the probability that an active actor activates its inactive neighbour, with each activation attempt occurring only once. We augment this model with an additional parameter δ , referred to as the *protocol function*, which governs how activation signals from multiple layers are aggregated to determine the overall state of an actor. When the protocol condition is satisfied, the actor (and all nodes representing it) becomes active; otherwise, the actor remains inactive, even if some of its layer-specific representations receive positive input. We consider two edge cases for δ : *AND*, where an actor is activated only if it receives positive input in *all* layers where it is represented; and *OR*, where activation in *any* layer is sufficient.

In total, we performed simulations under forty distinct configurations of MICM, with each experiment on a particular diffusion setup repeated forty times to reduce the stochastic effects inherent in the propagation model. Configurations of the MICM were defined by the Cartesian product of the two protocol functions: $\delta \in \{AND, OR\}$, and twenty activation probabilities: $\pi \in \{0.05, 0.10, \dots, 1.00\}$. It is worth noting that we adopted the cascade-based model, rather than LTM, as it allows diffusion to be initiated from a single actor — an essential requirement for computing individual spreading potentials.

3.3 Defining spreading-potential score

To enable ranking of actors according to Def. 3, we define a spreading-potential score that transforms the output vector $\mathbf{p}$ obtained during simulation into a unified scalar value.

Definition 8 (Spreading-potential score). *A score function that maps vectors of spreading potentials to a value in the range $[0, 1]$ is defined as:*

$$sps(\tilde{\mathbf{p}}) = \frac{1}{2}\tilde{p}_{ex} + \frac{1}{6}(1 - \tilde{p}_{sl}) + \frac{1}{6}\tilde{p}_{pi} + \frac{1}{6}(1 - \tilde{p}_{pl})$$

where:

$\tilde{\mathbf{p}}$ - the coordinate-wise normalised $\mathbf{p}$ by maximal value occurring across the network

The rationale behind this formulation reflects our interpretation of what constitutes a super-spreader. We prioritise the number of exposed actors, as it is typically regarded as the most critical metric in works related to spreading phenomena. All the remaining metrics are weighted equally, as none of them is deemed dominant. Moreover, our intention is to reward actors who spread information rapidly. Hence, we minimise $\tilde{p}_{sl}$ and $\tilde{p}_{pl}$ to favour fast and early spreaders. These weights, however, could also be arranged differently depending on the task. For instance, if one is concerned with abrupt diffusion, greater weight may be assigned to the activation peak ($\tilde{p}_{pi}$).

Once the score values were computed for all networks, we identified feasible ranges of MICM parameters to be considered in subsequent experiments. This step was necessary due to the inherent behaviour of spreading models, where only a limited subset of parameters yields “meaningful” diffusion — that is, processes that persist over several steps without either immediate extinction or full saturation. Finally, we determined these ranges separately for each protocol function, as their diffusion dynamics differ substantially. For $\delta = AND$, we selected $\pi \in \{0.80, 0.85, 0.90, 0.95\}$, while for $\delta = OR$, $\pi \in \{0.05, 0.10, 0.15, 0.20\}$. For each δ , the final *sps* value was obtained by averaging the vector $\mathbf{p}$ over the feasible values of π . Fig. 1 presents a typical distribution of the resulting scores.

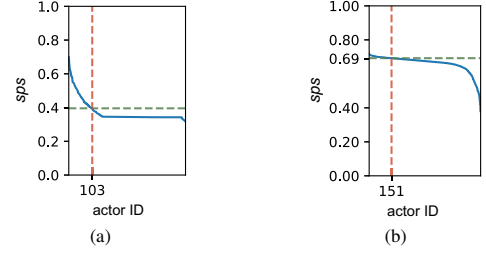


Figure 1: Distribution of *sps* in network-72 from artificial-er split, obtained via simulations under $\delta = AND$ (1a) and $\delta = OR$ (1b). Dashed lines indicate the cutoff for the most significant spreaders.

3.4 Final metrics used in the study

Following a detailed analysis of the *sps* distributions, we observed two distinct patterns. Under the $\delta = AND$ protocol, the distribution typically exhibits a discernible point separating actors with high spreading potential from those with limited ability to propagate information. In contrast, the $\delta = OR$ protocol results in a flatter curve, with the majority of actors displaying similar spreading capability and a sharp decline in *sps* values near the tail. Consequently, identifying top spreaders in the latter case becomes trivial, and we therefore excluded this data split from subsequent experiments.

This analysis led to the selection of the final metrics used to evaluate the performance of our proposed model and baseline methods. Primarily, we assessed the relative *sps* of the actor identified as the most influential spreader in the network. Additionally, to account for the stratified nature of the *sps* distribution, we defined a saddle ground-truth point s at the 80th percentile for each network, enabling the evaluation of prediction quality for top spreaders. To assess the robustness of the methods, we also considered the area under the cumulative relative score curve, both up to the saddle point and over the entire range. For clarity, these metrics are summarised in Tab. 2. Finally, we note a technical detail: to ensure consistency, all distributions were normalised by the number of actors (i.e., $k \in \{1, \dots, |A|\}$) was transformed into $\tilde{k} \in \{\frac{1}{|A|}, \dots, 1\}$), resulting in relative prediction curves confined to the $[0, 1]$ range on both axes.

Table 2: Final metrics, bounded within the range $[0, 1]$, used to assess the effectiveness of methods for identifying top spreaders.

Symbol	Formula	Description
T_{val}	$y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, \frac{1}{ A })$	Relative score of the most significant spreader identified
S_{auc}	$\sum_{\frac{1}{ A } \leq \tilde{k} \leq s} y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, \tilde{k})$	Area under the relative cumulated score curve up to the saddle point
S_{val}	$y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, s)$	Relative cumulated score of the identified top spreaders at the saddle point
F_{auc}	$\sum_{\frac{1}{ A } \leq \tilde{k} \leq 1} y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, \tilde{k})$	Area under the full relative cumulated score curve

4 Proposed model – TopSpreadersNetwork

One of the central contributions of this study is a supervised, inductive graph neural network model: *ts-net*. The model comprises approximately 458,000 trainable parameters and is designed to predict

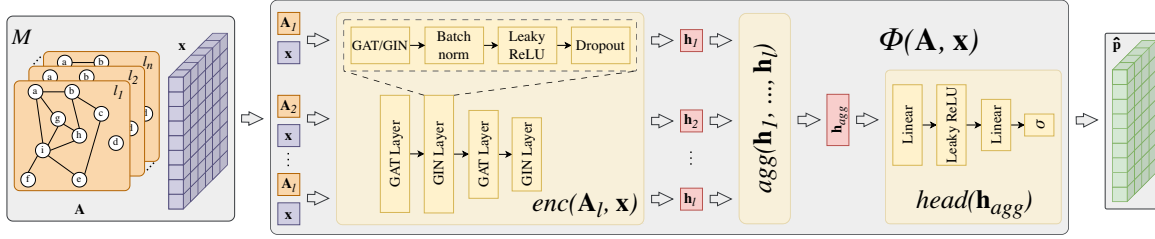


Figure 2: A schematic illustration of the *ts-net* architecture processing a multilayer network. Each layer is encoded independently using a shared encoder composed of interleaved GAT and GIN blocks. The resulting representations are then aggregated by a trainable aggregation layer to produce actor embeddings. Finally, a vector of spreading potential is generated using a component consisting of multilayer perceptrons.

a vector of spreading potentials $\hat{\mathbf{p}}$, which is subsequently used to identify super-spreaders in size-adaptive, multilayer networks. The core concept of the model is illustrated in Fig. 2. Owing to the adopted design choices, *ts-net* produces robust predictions across networks of varying size, density, and structural complexity.

4.1 Structural design

The *ts-net* is built upon three main components: (i) a shared encoder that produces actor embeddings for each layer of the network, (ii) a trainable aggregation module that combines these representations to derive final actor embeddings, and (iii) a prediction head that yields the output vector $\hat{\mathbf{p}}$ for each actor in the network. Each of these components is described in the following paragraphs.

Encoder. The encoder is shared across all layers L of the processed network M and is responsible for computing actor embeddings independently for each layer $l \in L$. It processes the layer-specific adjacency matrix $\mathbf{A}$ and a feature vector $\mathbf{x}$. This module consists of a four-layer hybrid GNN architecture that combines GIN [49] and GAT [44] blocks, enhanced with dropout [18], batch normalisation [19], and LeakyReLU [32] activation. The hybrid design enables the model to capture rich structural context while maintaining strong generalisation capabilities, particularly in networks with complex topologies, aligning with the intuition presented in [35] and supported by the experimental results (see App. A).

Definition 9 (*WiseAverage fusion*). The trainable aggregation mechanism that computes the actor embedding matrix $\mathbf{h}_{agg}$ from a collection of actor representations across network layers $\{\mathbf{h}_1, \dots, \mathbf{h}_{|L|}\}$, is defined as follows:

$$\mathbf{h}_{agg} = \sum_{l=1}^{|L|} \mathbf{T}^{(l)} \odot \mathbf{h}_l, \quad \mathbf{T} = \text{softmax} \left(\begin{bmatrix} \mathbf{W}\mathbf{h}_1 \\ \mathbf{W}\mathbf{h}_2 \\ \vdots \\ \mathbf{W}\mathbf{h}_{|L|} \end{bmatrix} \right)$$

$$\mathbf{h}_{agg} \in \mathbb{R}^{|A| \times d}, \mathbf{T} \in \mathbb{R}^{|L| \times |A| \times 1}, \mathbf{h}_l \in \mathbb{R}^{|A| \times d}, \mathbf{W} \in \mathbb{R}^{1 \times d}$$

where:

- $\mathbf{h}_{agg}$ - the final aggregated embedding matrix,
- $\mathbf{T}$ - the attention matrix computed via softmax,
- $\mathbf{h}_l$ - the embedding from the l -th layer of the processed network,
- $\mathbf{W}$ - a trainable weight matrix from the linear layer,
- d - the embedding dimension,
- $|A|$ - the number of actors in the processed network,
- $|L|$ - the number of layers in the processed network.

Aggregation module. To combine actor representations from each layer of the processed network, we utilise a *soft attention* mechanism [1]. Our approach — *WiseAverage* (Def. 9) is motivated by the intuition that an actor's relative importance may vary across layers; hence, aggregating layer-specific embeddings should be performed adaptively. Accordingly, each actor is assigned a normalised vector that reflects its importance on each layer and is then used to compute a weighted average of the layer-specific embeddings, yielding its network-level representation. As shown in the experimental part (see App. A), this fusion mechanism consistently outperforms several well-established embedding-level aggregation methods [2].

Prediction head. The final component is built upon a two-layer MLP that predicts the vector $\mathbf{p}$ of spreading potentials (Def. 7), which is then used to compute scores and derive the ranking $\hat{\mathbf{R}}$.

4.2 Techniques for robust prediction

This section outlines the key design choices made to improve *ts-net*'s predictive capabilities. Fig. 3 presents the data processing flow during training and inference. Although the former largely follows standard supervised learning, several non-trivial adaptations were introduced to improve model performance and generalisation.

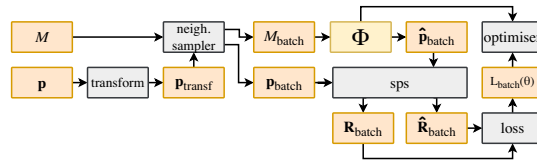


Figure 3: The pipeline used to train the *ts-net*; orange blocks represent data entities, grey blocks indicate functional components, and the model is highlighted in yellow. Note that while the model is trained to optimise $\hat{\mathbf{R}}$, it is still capable of predicting $\mathbf{p}$.

Data transformations. Since the employed diffusion mechanism considers solely actor's connections across the network, no other actor attributes are meaningful in the context of this problem. Therefore, we use zero vectors as actor features, i.e., $\mathbf{x} = \mathbf{0} \in \mathbb{R}^{|A| \times d_{in}}$. As shown in App. A, this choice is supported by experiments involving alternative feature sets. Moreover, motivated by insights from the dataset analysis (Sec. 3.4), we introduced a data transformation module into the training pipeline to enhance the visibility of super-spreaders. Specifically, we applied a function *transform* : $\mathbf{p} \rightarrow \mathbf{p}_{transf}$ to the ground-truth labels used during training. After evaluating the effectiveness of various approaches (see App. A),

we selected one referred to as *scatter*, which amplifies the scores of high-potential spreaders while attenuating those of weaker ones. The transformation is defined as $\mathbf{p}_{transf} = \exp(3 \cdot \mathbf{p}) / \exp(3)$. Importantly, this function preserves interpretability, as it is invertible and allows for reconstruction of the original target values.

Local predictions. To maximise the throughput of the *ts-net* when processing large and complex networks, we adopted the sampling technique introduced in GraphSAGE [16] (implemented in PyG [14] as the `NeighborLoader` class). The configuration was adjusted to ensure that only larger networks are subsampled, while smaller graphs are processed in full. This enables the model to learn from local rankings, significantly reducing computational requirements while allowing it to focus on the most relevant substructures within the network. As a result, prediction quality can be improved through better alignment with local topological patterns [7].

Optimisation task. For the training task, we employed the ListMLE loss function [48], which is designed to optimise the model by sorting predicted scores. This enables the model to learn local rankings derived from neighbourhood samples. Importantly, although the model is trained to predict rankings $\hat{\mathbf{R}}$, it outputs vectors of spreading potentials $\hat{\mathbf{p}}$. This feature enhances the model’s interpretability, a characteristic often lacking in competitive methods, which typically do not provide explanations for their predictions.

5 Evaluation and results

A comprehensive set of experiments was designed to evaluate the effectiveness of *ts-net* with respect to the following questions: (i) how does the proposed method perform on both synthetic and real-world multilayer networks compared to competitive solutions (ii) can an inductive approach achieve performance comparable to that of state-of-the-art approaches (iii) what are the limitations of the examined methods in handling structurally diverse graphs (iv) how do the individual components of *ts-net* contribute to its overall performance (v) how do different hyperparameter settings, tasks, and dataset variants affect the performance of *ts-net*? In the following section, we provide answers to most of these questions, while referring interested readers to App. A for a detailed discussion of the latter two.

5.1 Experimental setup

To provide a comprehensive comparison of the capabilities of *ts-net*, we selected seven competitive solutions, encompassing both heuristic methods and learning-based models. Notably, all of them address the broader problem of influence maximisation, rather than the more specific task of identifying top spreaders. Moreover, some were originally developed for homogeneous networks and thus required adaptation to the multilayer context. Our choice was intended to reflect a broad spectrum of successful, state-of-the-art methods that are most closely related in objective and scope to the considered problem.

The heuristic group includes classic centrality measures: Degree Centrality (*deg-c*) and One-Hop Neighbourhood Size [33] (*nghb-s*), along with their discounting counterparts: Degree Centrality Discount [6] (*deg-cd*) and Neighbourhood Size Discount [9] (*nghb-sd*). These were chosen for their low computational cost and ability to yield actor rankings compatible with multilayer settings. We also include a random baseline for reference. For learning-based models, we selected DeepIM [30] due to its data-driven design and recent impact in influence maximisation on homogeneous graphs. Despite its transductive nature, it serves as a strong neural baseline. As a representative of unsupervised embedding-based techniques adaptable to

multilayer data, we selected Multi-node2vec [46] with KMeans clustering (*mn2v-km*) to represent unsupervised embedding-based techniques adaptable to multilayer data.

The experimental procedure consists of three stages. First, *ts-net*, being the only inductive method among those evaluated, is trained on synthetic graphs, with 168 used for training and 20 for validation. In the second stage, all methods are applied to generate rankings of super spreaders on both synthetic and real-world networks not seen by the *ts-net*. For transductive methods, training is performed individually on each test network at this stage. Finally, the resulting rankings are evaluated using the metrics listed in Tab. 2. We refer the reader to App. B for results based on standard ranking metrics, which, however, are not substantially different from those presented below.

5.2 Result analysis

We begin the discussion of results by analysing the average performance of the evaluated methods, as illustrated by the cumulative relative score curves in Fig. 4 and the corresponding metrics in Tab. 3. *ts-net* achieved the best results in 7 out of 8 scenarios, significantly outperforming all other methods, particularly on real-world networks under the T_{val} metric, which captures the ability to identify the most influential spreader for the entire graph. The only case in which *ts-net* was not ranked first was for real networks under the S_{val} metric, where *deg-cd* held a slight advantage (by 0.004). Among the baselines, only *deg-cd* and *deg-c* delivered comparable results, while all other methods exhibited substantially lower performance. These findings are further supported by Fig. 4, which also reveals a marked drop in performance across most methods beyond the top-ranked actors, indicating room for further improvement and investigation.

Table 3: Average performance of the evaluated methods across two network splits. Predictions with the highest quality (i.e., closest to 1) for each criterion are highlighted in green, with darker shades indicating better ranks.

ϕ	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
<i>random</i>	0.538	0.672	0.759	0.875	0.520	0.629	0.749	0.863
<i>deg-c</i>	0.841	0.821	0.872	0.942	0.829	0.819	0.896	0.931
<i>deg-cd</i>	0.830	0.820	0.872	0.943	0.829	0.815	0.901	0.933
<i>nghb-s</i>	0.706	0.772	0.834	0.924	0.777	0.806	0.887	0.918
<i>nghb-sd</i>	0.662	0.771	0.830	0.920	0.777	0.801	0.886	0.919
<i>deep-im</i>	0.548	0.679	0.761	0.878	0.643	0.677	0.788	0.872
<i>mn2v-km</i>	0.611	0.711	0.790	0.903	0.586	0.658	0.786	0.881
<i>ts-net</i>	0.850	0.826	0.881	0.947	0.921	0.826	0.897	0.934

A crucial aspect of GNNs is their ability to generalise. To examine this property, a detailed analysis of the performance on real-world networks, not seen by *ts-net* during training, is presented in Tab. 4.

Despite this setup, *ts-net* was the only method to consistently achieve scores no lower than 0.790 across all metrics and datasets. Moreover, it attained the best performance in 11 out of the 21 evaluation cases, outperforming all other methods; the second-best method, *deg-cd*, was ranked first in 9 cases. *ts-net* obtained the highest T_{val} score on five out of eight networks and consistently ranked among the top three in the remaining metrics. Its performance was especially pronounced on *timik*, demonstrating that predicting local rankings to improve memory efficiency does not compromise predictive quality. Notable performance deviations were observed on the *arxiv* and *eu-trans* networks, both having considerably more layers (13 and 37, respectively) than the maximum of 5 in the training data. The performance drop on out-of-distribution networks may stem from lim-

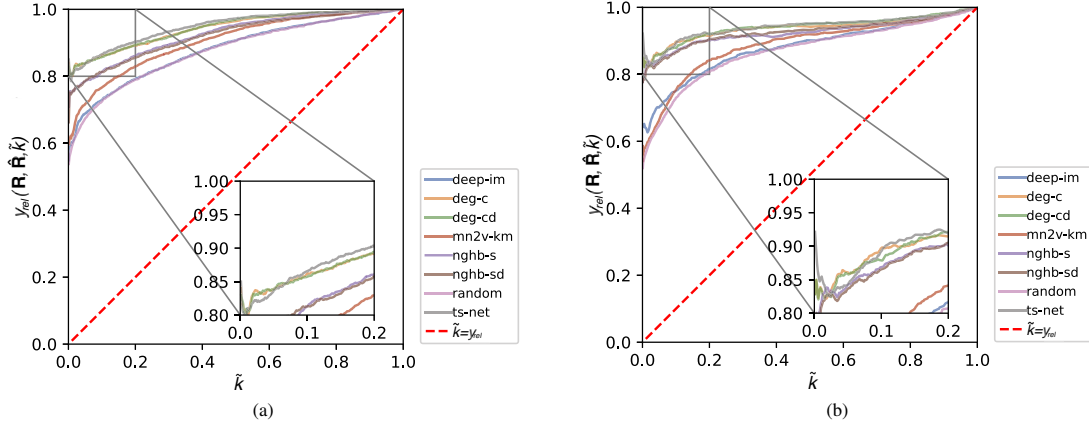


Figure 4: Averaged cumulated relative score curves from prediction on test artificial (4a) and real-world (4b) networks by the evaluated methods.

Table 4: Performance of the evaluated methods on the real-world networks. Predictions with the highest quality (i.e., closest to 1) for each criterion are highlighted in green, with darker shades indicating better ranks. S_{auc} covers mostly with S_{val} and is hidden for conciseness.

ϕ	arxiv			aucs			ckmp			eu-trans			l2-course			lazega			timik		
	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}	T_{val}	S_{val}	F_{auc}
random	0.390	0.810	0.920	0.640	0.810	0.876	0.460	0.790	0.878	0.420	0.750	0.887	0.740	0.840	0.859	0.660	0.750	0.809	0.330	0.490	0.813
deg-c	1.000	0.930	0.956	0.990	0.940	0.953	0.980	0.830	0.922	0.710	0.890	0.936	0.870	0.930	0.920	0.950	0.930	0.934	0.300	0.820	0.892
deg-cd	1.000	0.920	0.953	0.990	0.940	0.957	0.980	0.860	0.923	0.710	0.900	0.944	0.870	0.920	0.916	0.950	0.950	0.941	0.300	0.820	0.894
ngbh-s	1.000	0.940	0.961	0.640	0.920	0.935	0.980	0.810	0.901	0.700	0.900	0.937	0.870	0.890	0.891	0.950	0.930	0.910	0.300	0.820	0.889
ngbh-sd	1.000	0.930	0.955	0.640	0.930	0.939	0.980	0.810	0.905	0.700	0.900	0.943	0.870	0.890	0.893	0.950	0.940	0.911	0.300	0.800	0.888
deep-im	0.400	0.810	0.921	0.570	0.760	0.855	0.460	0.800	0.885	0.680	0.760	0.895	0.880	0.800	0.849	0.870	0.800	0.829	OOM	OOM	OOM
mn2v-km	0.520	0.820	0.924	0.550	0.770	0.866	0.480	0.880	0.927	0.400	0.740	0.882	0.800	0.820	0.842	0.390	0.870	0.859	0.960	0.600	0.868
ts-net	0.880	0.890	0.952	1.000	0.940	0.947	0.980	0.860	0.927	0.790	0.880	0.936	0.870	0.930	0.921	0.950	0.930	0.938	0.980	0.850	0.917

itations of the aggregation component. While scalability w.r.t. the number of actors has been achieved using `NeighborLoader`, the number of relationships requires further investigation. The observed effect similar to that of oversquashing (too much information from many layers is compressed into one vector) could be mitigated using techniques analogous to those employed for handling large neighbourhoods; i.e., by processing the layers “chunk by chunk”, rather than all at once. Naturally, enriching the dataset with networks containing more layers may also help.

Among the other methods, *deg-cd* and *deg-c* yielded stable and competitive performance, especially on smaller networks, where they frequently ranked in the top three. However, they underperformed on more irregular graphs such as *timik*. The results of *ngbh-s* and *ngbh-sd* were mixed: while effective in T_{val} on *arxiv*, *ckmp*, and *lazega*, their scores in S_{val} and F_{auc} were inconsistent. The *random*, predictably, placed last in most cases. *mn2v-km* performed reasonably on *ckmp* and *timik*, but fell behind top methods overall. Finally, *DeepIM* performed poorly, except for T_{val} on the smallest *l2-course*, and failed to run on the largest graph (*timik*) due to memory issues.

6 Conclusions

This study addresses the problem of identifying super-spreaders in multilayer networks by proposing a novel solution — *TopSpreaderNetwork* — an inductive graph neural network capable of processing multilayer networks of varying sizes and complexities. To this end, we also constructed the *TopSpreadersDataset*, which captures actor-level diffusion characteristics under the Multilayer Independent Cascade Model. Finally, we proposed and validated alternative techniques for data transformation and embedding fusion that improve

the robustness of ranking-based learning from multirelational data.

Nonetheless, some limitations remain. While multilayer networks constitute a powerful and flexible framework, they are still a subset of the broader class of heterogeneous graphs, which are not directly addressed in this work. Moreover, although *TopSpreaderNetwork* demonstrates superior performance, relatively simple heuristics such as Degree Centrality continue to yield strong results (for a comparison of time performance see App. C). The main advantage of our approach lies in its interpretability: rather than producing scores in a black-box fashion, it first predicts a vector of four interpretable features, which can then be used to construct tailored rankings. This offers flexibility for use cases where, e.g., actors causing the highest peak of infections are more desirable than those triggering longer diffusion, something other solutions do not easily allow.

Several directions exist for future research. One interesting option is to include diffusion results for small yet effective seed sets, e.g., generated using the greedy algorithm, thereby extending the learning task from scoring individual actors to evaluating entire sets. Another promising avenue is to reforge the dataset into a general benchmark for super-spreader identification, encompassing a broader range of diffusion models and network topologies.

Acknowledgements

This research was partially supported by the National Science Centre, Poland, grant no. 2022/45/B/ST6/04145, the Polish Ministry of Science and Higher Education programme: International Projects Co-Funded, and the EU under the Horizon Europe, grant no. 101086321. Views and opinions expressed here are those of the authors only and do not necessarily reflect those of the funding agencies.

References

- [1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint:1409.0473*, 2014.
- [2] P. Bielak and T. Kajdanowicz. Representation learning in multiplex graphs: Where and how to fuse information? In *International Conference on Computational Science*, pages 3–18. Springer, 2024.
- [3] B. Bollobas, C. Borgs, J. Chayes, and O. Riordan. Directed scale-free graphs. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 132–139, 1 2003.
- [4] A. Cardillo, J. Gómez-Gardeñes, M. Zanin, M. Romance, D. Papo, F. d. Pozo, and S. Boccaletti. Emergence of network features from multiplexity. *Scientific Reports*, 3(1344):1–6, 2 2013. ISSN 2045-2322.
- [5] B. Chamberlain, J. Rowbottom, M. I. Gorinova, M. Bronstein, S. Webb, and E. Rossi. Grand: Graph neural diffusion. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1407–1418. PMLR, 18–24 Jul 2021.
- [6] W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conf. on Knowledge discovery and data mining*, pages 199–208, 2009.
- [7] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, and C.-J. Hsieh. Clustergcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD international conf. on knowledge discovery and data mining*, pages 257–266, 2019.
- [8] J. Coleman, E. Katz, and H. Menzel. The diffusion of an innovation among physicians. *Sociometry*, 20(4):253–270, 1957. ISSN 00380431.
- [9] M. Czuba and P. Bródka. Rank-refining seed selection methods for budget constrained influence maximisation in multilayer networks under linear threshold model. *Social Network Analysis and Mining*, 15(1): 46–25, Apr. 2025. ISSN 1869-5469.
- [10] M. Czuba, M. Stolarski, A. Piróg, P. Bielak, and P. Bródka. Identifying super spreaders in multilayer networks [preprint with appendices]. *arXiv*, May 2025. doi: 10.48550/arXiv.2505.20980.
- [11] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X*, 5:011027, 3 2015.
- [12] X. Deng, F. Long, B. Li, D. Cao, and Y. Pan. An influence model based on heterogeneous online social network for influence maximization. *IEEE Transactions on Network Science and Eng.*, 7:737–749, 2020.
- [13] P. Erdős, A. Rényi, et al. On the evolution of random graphs. *Publ. math. inst. hung. acad. sci.*, 5(1):17–60, 1960.
- [14] M. Fey and J. E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.
- [15] J. Goldenberg, B. Libai, and E. Muller. Talk of the Network: A Complex Systems Look at the Underlying Process of Word-of-Mouth. *Marketing Letters*, 12(3):211–223, Aug. 2001. ISSN 1573-059X.
- [16] W. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. *Advances in neural inf. processing sys.*, 30, 2017.
- [17] Y. He, Q. Gan, D. Wipf, G. D. Reinert, J. Yan, and M. Cucuringu. Gnnrank: Learning global rankings from pairwise comparisons via directed gnns. In *international conference on machine learning*, 2022.
- [18] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
- [19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
- [20] J. Jankowski, R. Michalski, and P. Bródka. A multilayer network dataset of interaction and influence spreading in a virtual world. *Scientific Data*, 4(1):170144, 2017. ISSN 2052-4463.
- [21] V. Rao, K. M. Katukuri, and M. Jagarapu. CIM: clique-based heuristic for finding influential nodes in multilayer networks. *Applied Intelligence*, 52(5):5173–5184, Mar. 2022. ISSN 1573-7497.
- [22] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.
- [23] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks, 2017.
- [24] M. Kivelä, A. Arenas, M. Barthélemy, J. P. Gleeson, Y. Moreno, and M. A. Porter. Multilayer networks. *Journal of Complex Networks*, 2(3): 203–271, 07 2014. ISSN 2051-1310.
- [25] S. Kumar, A. Mallik, A. Khetarpal, and B. Panda. Influence maximization in social networks using graph embedding and graph neural network. *Information Sciences*, 607:1617–1636, 2022. ISSN 0020-0255.
- [26] Y. Li, L. Li, Y. Liu, and Q. Li. Mahe-im: Multiple aggregation of heterogeneous relation embedding for influence max. on heterogeneous information network. *Expert Sys. with Appl.*, 202:117289, 04 2022.
- [27] C. Ling, T. Chowdhury, J. Jiang, J. Wang, X. Zhang, H. Chen, and L. Zhao. Deepgar: Deep graph learning for analogical reasoning, 2022.
- [28] C. Ling, J. Jiang, J. Wang, and Z. Liang. Source localization of graph diffusion via variational autoencoders for graph inverse problems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD ’22, page 1010–1020. ACM, Aug. 2022.
- [29] C. Ling, H. Cao, and L. Zhao. STGEN: Deep Continuous-Time Spatiotemporal Graph Generation. In *Machine Learning and Knowledge Discovery in Databases*, pages 340–356. Springer, Cham, Switzerland, Mar. 2023. ISBN 978-3-031-26409-2.
- [30] C. Ling, J. Jiang, J. Wang, M. T. Thai, L. Xue, J. Song, M. Qiu, and L. Zhao. Deep graph representation learning and optimization for influence maximization. In *Proceedings of the 40th International Conference on Machine Learning*, ICML’23. JMLR.org, 2023. doi: 10.5555/3618408.3619288.
- [31] Q. Liu, R. Xie, L. Chen, S. Liu, K. Tu, P. Cui, B. Zhang, and L. Lin. Graph neural network for tag ranking in tag-enhanced video recommendation. In *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020.
- [32] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013. ICML Workshop on Deep Learning for Audio, Speech and Language Processing.
- [33] M. Magnani and L. Rossi. The ml-model for multi-layer social networks. In *2011 International conference on advances in social networks analysis and mining*, pages 5–12. IEEE, 2011.
- [34] M. Magnani, L. Rossi, and D. Vega. Analysis of multiplex social networks with r. *Journal of Statistical Software*, 98(8):1–30, 2021.
- [35] C.-H. Pan, Y. Qu, Y. Yao, M.-J.-S. Wang, et al. Hybridgnn: A self-supervised graph neural network for efficient maximum matching in bipartite graphs. *Symmetry*, 16(12):1631, 2024.
- [36] M. B. Paradowski, N. Whitby, M. Czuba, and P. Bródka. Peer Interaction Dynamics and L2 Learning Trajectories During Study Abroad: A Longitudinal Investigation Using Dynamic Computational Social Network Analysis. *Language Learning*, 74(S2):58–115, Dec. 2024.
- [37] L. Rossi and M. Magnani. Towards effective visual analytics on multiplex and multilayer networks. *Chaos, Solitons & Fractals*, 72:68–76, 2015. ISSN 0960-0779.
- [38] M. Salehi, R. Sharma, M. Marzolla, M. Magnani, P. Siyari, and D. Montesi. Spreading processes in multilayer networks. *IEEE Transactions on Network Science and Engineering*, 2(2):65–83, 2015.
- [39] C. Shi, X. Wang, and P. S. Yu. *Introduction*, pages 1–8. Springer Singapore, Singapore, 2022. ISBN 978-981-16-6166-2.
- [40] S. S. Singh, D. Srivastava, M. Verma, and J. Singh. Influence maximization frameworks, performance, challenges and directions on social network: A theoretical study. *Journal of King Saud University-Computer and Information Sciences*, 34(9):7570–7603, 2022.
- [41] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock. New specifications for exponential random graph models. *Sociological Methodology*, 36(1):99–153, 2006.
- [42] M. Stolarski, A. Piróg, and P. Bródka. Identifying key nodes for the influence spread using a ml approach. *Entropy*, 26(11):955, 2024.
- [43] S. Tian, S. Mo, L. Wang, and Z. Peng. Deep Reinforcement Learning-Based Approach to Tackle Topic-Aware Influence Maximization. *Data Science and Engineering*, 5(1):1–11, Mar. 2020. ISSN 2364-1541.
- [44] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.
- [45] J. Wang, J. Jiang, and L. Zhao. An invertible graph diffusion neural network for source localization, 2022.
- [46] J. D. Wilson, M. Baybay, R. Sankar, P. Stillman, and A. M. Popa. Analysis of population functional connectivity data via multilayer network embeddings. *Network Science*, 9(1):99–122, 2021.
- [47] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2021.
- [48] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199, 2008.
- [49] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.
- [50] Z. Yuan, M. Shao, and Z. Chen. Graph Bayesian Optimization for Multiplex Influence Maximization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Mar. 2024.
- [51] Y. D. Zhong, V. Srivastava, and N. E. Leonard. Influence spread in the heterogeneous multiplex linear threshold model. *IEEE Transactions on Control of Network Systems*, 9(3):1080–1091, 2022.

A Model selection and ablation study

In this section, we present experiments on the *ts-net*, exploring various architectural designs, training configurations, and dataset settings. As illustrated in Fig. 3, the training pipeline comprises several components, each of which could be examined in detail. However, to maintain conciseness, we report only the most essential aspects.

In each of the following subsections, the baseline architecture of the *ts-net* remains unchanged. Specifically, we employ an encoder composed of a sequence of GAT-GIN-GAT-GIN neural layers, with a maximum hidden dimensionality of 400 channels. Each layer is followed by batch normalisation, a LeakyReLU activation, and dropout. For the aggregation layer, we use the *WiseAverage* (avg_w) block, whereas vector of zeros serves as a feature matrix $\mathbf{x}$ and the spreading-potential scores are normalised columnwise by a maximal value. This design was selected on the basis that it proved most effective in preliminary experiments. In each table, we then present how the model’s performance is affected when a specific parameter of the model or the training pipeline is modified.

A.1 Model

We begin with an ablation study of the model itself. Following the data flow direction, we evaluate the architectural designs of the encoder layers (Tab. 5), the number of hidden channels in the encoder (Tab. 6), and the aggregation layers (Tab. 7).

Table 5: Performance of *ts-net* with encoder’s architectures; S_{auc} covers mostly with S_{val} and is hidden for conciseness.

Neural layers	Norm. and activat.	Artificial networks			Real networks		
		T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}
2x(GAT,GIN)	BN,LReLU,DRPT	0.808	0.884	0.947	0.899	0.899	0.929
4xGIN	BN,LReLU,DRPT	0.657	0.810	0.898	0.849	0.850	0.914
4xGAT	BN,LReLU,DRPT	0.489	0.761	0.877	0.569	0.780	0.881
4xGCN	BN,LReLU,DRPT	0.510	0.769	0.885	0.549	0.776	0.884
2x(GAT,GIN)	BN,LReLU	0.816	0.878	0.946	0.864	0.871	0.923
2x(GAT,GIN)	BN,RELU,DRPT	0.499	0.696	0.822	0.480	0.690	0.825
2x(GAT,GIN)	LReLU,DRPT	0.846	0.888	0.950	0.839	0.866	0.919
2x(GAT,GIN)	RELU	0.827	0.883	0.949	0.863	0.870	0.921
3xGIN	RELU	0.633	0.821	0.910	0.661	0.820	0.894

Experiments covering various encoder designs (Tab. 5) are divided into three groups. First, we attempt to replace the interleaved layers with a single type repeated four times, preserving the model’s depth. Second, we simplify the normalisation and activation components that follow each neural layer. Finally, we test a substantially simplified encoder with reductions in both aspects. This experiment is aimed to support our unconventional choice of interleaving different types of neural layers.

As shown, removing the interleaved layers generally leads to a drop in performance across all metrics. We hypothesise that the interplay between GAT layers, which prioritise influential neighbourhoods, and GIN layers, which are effective at distinguishing topological structures, reflects the MICM most accurately, thereby enhancing performance. When simplifying the normalisation and activation components, we observe a tendency of the model to overfit to artificial networks drawn from the same distribution as the training data. This naturally leads to poorer generalisation on real-world networks. The smallest drop in performance was observed when dropout alone was removed or when LeakyReLU was replaced with standard ReLU. Finally, we also evaluated a highly simplified encoder consisting of three GIN layers followed by ReLU activations. This configuration showed a substantial performance decline across all metrics,

although not the biggest which was observed for encoder build with 4 GCN layers.

Table 6: Performance of *ts-net* with different hidden dimensionalities.

d_{hidden}	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
200	0.619	0.729	0.821	0.909	0.593	0.645	0.753	0.863
400	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
600	0.796	0.806	0.861	0.931	0.886	0.794	0.877	0.925
800	0.784	0.806	0.867	0.944	0.856	0.797	0.884	0.924

With regards to the next evaluated property of the model, it achieves the best performance with 400 hidden channels. We observe that 200 channels are insufficient, as the model performs worst in this setting. While 600 and 800 channels yield results comparable to 400, they do not outperform it on any metric. Consequently, we adopt 400 as the optimal size.

Table 7: Performance of *ts-net* with different aggregation layers.

$aggr(\cdot)$	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
<i>min</i>	0.658	0.750	0.808	0.903	0.721	0.705	0.814	0.905
<i>max</i>	0.678	0.803	0.873	0.942	0.743	0.750	0.851	0.915
<i>avg</i>	0.721	0.787	0.859	0.940	0.799	0.737	0.841	0.919
<i>avg_w</i>	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
<i>sum</i>	0.655	0.723	0.799	0.897	0.679	0.692	0.806	0.888
<i>att</i>	0.711	0.785	0.846	0.930	0.826	0.745	0.847	0.911

The final architectural experiments (Tab.7) focused on the aggregation layer. The highest performance was obtained with the *avg_w* block proposed by us, followed by the attention-based (*att*) and average (*avg*) aggregators. This outcome supports our intuition: the aggregation mechanism should combine the smoothing properties of averaging with the ability to focus on the most important network’s layers.

A.2 Task

By conducting this series of experiments, we aimed to evaluate the model’s performance when trained using two loss functions suitable for the considered task. We also sought to examine the expressiveness of the dataset and assess whether training on a data split constructed using spreading outcomes from one regime is sufficient to accurately predict spreading potentials computed under another one.

Table 8: Performance of *ts-net* on the considered task under different training loss functions.

$\mathcal{L}$	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
ListMLE	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
WMAE	0.758	0.801	0.871	0.938	0.887	0.789	0.867	0.919

In the experiment reported in Tab. 8, we compare two loss functions. The first, ListMLE, is designed for ranking prediction tasks. The second, Weighted Mean Absolute Error (WMAE), is primarily used in regression problems and is conceptually close to the task under consideration. As shown, the use of ListMLE consistently yields better performance across all evaluation criteria. Nonetheless, it is worth noting that WMAE does not perform significantly worse.

We also investigated the dataset’s expressiveness in terms of the model’s ability to generalise. To this end, we trained the *ts-net* on spreading potentials generated using the MICM with one protocol

Table 9: Expressiveness of *TopSpreadersDataset* split by δ , as reflected in the generalisation performance of *ts-net*.

Train δ	Test δ	Artificial networks				Real networks			
		T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
AND	AND	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
OR	AND	0.743	0.748	0.805	0.901	0.850	0.761	0.854	0.915
AND	OR	0.982	0.979	0.993	0.994	0.940	0.912	0.946	0.954
OR	OR	0.955	0.957	0.976	0.982	0.924	0.911	0.949	0.956

function (δ), and tested it on data from simulations executed with a different protocol. The results are summarised in Tab. 9 and should be interpreted in reference to Fig. 1. As spreading potentials tend to be more evenly distributed when computed from simulations based on the *OR* protocol, the model trained on such data exhibits inferior generalisation compared to the one trained on data obtained with $\delta = \text{AND}$, where the distributions are sharper and super-spreaders are more clearly distinguishable. This supports the claim that identifying super-spreaders using data generated under the *OR* protocol may be regarded as a trivial task, and therefore unworthy of further investigation.

A.3 Data transformations

The final part of the ablation study concerns dataset processing setups. Here, we examine which types of actor features result in the best predictive performance of the model and how transforming the spreading potentials affects this outcome.

Table 10: Performance of *ts-net* when trained on different feature vectors.

x	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
zeros	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
ones	0.668	0.702	0.768	0.874	0.707	0.704	0.791	0.881
central.	0.764	0.801	0.858	0.934	0.730	0.766	0.869	0.926

Tab. 10 presents the impact of different types of actor features. We evaluate three configurations: five-dimensional vectors of ones, five-dimensional zero vectors, and vectors composed of five centrality measures: degree, betweenness, closeness, core number, neighbourhood size — along with VoteRank. All centralities were computed using formulations designed for multilayer networks. Given the nature of the task, we restrict the analysis to these feature sets, since the MICM accounts solely for an actor’s position within the network and no other actor attributes are meaningful in the context of this problem. As shown, the best predictive power is achieved when the model is trained on zero-vectors. This also offers a practical advantage of our proposed approach: to identify super-spreaders, only the network structure is required — incorporating centrality measures does not improve the model’s effectiveness.

The final experiment, albeit one of the most significant, was aimed at identifying the most effective data transformation for the dataset to modify the values of the spreading-potential vector $\mathbf{p}$ (Def. 7) in order to facilitate the ranking prediction task. We evaluated seven transformation operators, along with a model trained without any them.

The $\text{norm}_{\max}$ operator denotes max-normalisation, while norm_{act} corresponds to normalisation by the number of actors in the network — both applied independently to each coordinate of the vector. The $\text{norm}_{act,diam}$ operator applies a combined scheme: p_{ex} and p_{pi} are normalised by the number of actors, whereas p_{sl} and p_{pl} are normalised by the network’s diameter. Next, the $\log$ transforma-

tion is defined as $\mathbf{p}_{transf} = \log(\mathbf{p} + 1)$. The *scatter* operator is designed to emphasise the values of actors with high spreading potential while diminishing those of weak spreaders. It is defined as $\mathbf{p}_{transf} = \exp(3 \cdot \mathbf{p}) / \exp(3)$. Finally, we also consider composite transformations, namely $\log \circ \text{norm}_{\max}$ and $\text{scatter} \circ \text{norm}_{\max}$, which apply the corresponding operators in sequence.

Table 11: Performance of *ts-net* with different data transformations applied.

$\text{transf}(\cdot)$	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
none	0.644	0.764	0.845	0.928	0.841	0.772	0.867	0.915
$\text{norm}_{\max}$	0.808	0.831	0.884	0.947	0.899	0.814	0.899	0.929
norm_{act}	0.648	0.725	0.807	0.902	0.811	0.714	0.810	0.905
$\text{norm}_{act,diam}$	0.762	0.806	0.867	0.938	0.893	0.810	0.894	0.930
$\log$	0.631	0.735	0.816	0.908	0.800	0.698	0.800	0.899
$\log \circ \text{norm}_{\max}$	0.648	0.711	0.786	0.892	0.731	0.712	0.801	0.892
<i>scatter</i>	0.850	0.826	0.881	0.947	0.921	0.826	0.897	0.934
$\text{norm}_{\max} \circ \text{scatter}$	0.633	0.749	0.832	0.919	0.644	0.744	0.857	0.905

As shown in Tab. 11, two transformations yielded the best results: $\text{norm}_{\max}$ and *scatter*. However, due to its superior generalisation ability on real-world networks and the possibility of inverting the transformation, we select the latter for the final comparison with competitive methods. Among the remaining operators, $\text{norm}_{act,diam}$ also achieved solid performance, with the additional advantage of enabling the recovery of non-relative values for the predicted spreading-potential vector.

B Evaluation on other metrics

In addition to evaluating the proposed model against competitive baselines using the relative cumulative score (Def. 5), we also employed metrics commonly used in the domain of ranking-learning. Instead of assessing prediction quality based on the scores assigned to top- k ranked actors, these metrics evaluate only the alignment between the indices in $\hat{\mathbf{R}}$ and $\mathbf{R}$. Specifically, we report:

- Precision — the fraction of correctly predicted actors in $\hat{\mathbf{R}}$;
- Jaccard index — the overlap between the sets $\hat{\mathbf{R}}$ and $\mathbf{R}$.

We report these metrics, similarly to the main metrics used in the study (Tab. 2), at three levels of the ranking: at its top position (T), at the cut-off separating super-spreaders (S), as illustrated in Fig. 1, and over the full ranking (F). For each metric, we report the average score computed across all prefixes of the ranking, that is, over all top- k sets for $k = 1, 2, \dots, \{T, S, F\}$.

Table 12: Average **precision** of the evaluated methods across two network splits. Predictions with the highest quality (~ 1) are highlighted in green, with darker shades indicating better ranks.

ϕ	Artificial networks			Real networks		
	T	S	F	T	S	F
random	0.004	0.087	0.502	0.010	0.092	0.505
deg-c	0.250	0.370	0.629	0.143	0.426	0.612
deg-cd	0.250	0.370	0.631	0.143	0.427	0.621
nghb-s	0.062	0.250	0.582	0.143	0.394	0.579
nghb-sd	0.000	0.249	0.577	0.143	0.382	0.587
deep-im	0.000	0.098	0.506	0.000	0.095	0.497
mn2v-km	0.000	0.147	0.547	0.000	0.138	0.533
ts-net	0.250	0.382	0.638	0.143	0.450	0.624

Table 13: Average **Jaccard** index of the evaluated methods across two network splits. Predictions with the highest quality (~ 1) are highlighted in green, with darker shades indicating better ranks.

ϕ	Artificial networks			Real networks		
	<i>T</i>	<i>S</i>	<i>F</i>	<i>T</i>	<i>S</i>	<i>F</i>
<i>random</i>	0.004	0.047	0.389	0.010	0.053	0.392
<i>deg-c</i>	0.250	0.242	0.493	0.143	0.289	0.477
<i>deg-cd</i>	0.250	0.241	0.494	0.143	0.289	0.488
<i>nghb-s</i>	0.062	0.150	0.451	0.143	0.263	0.445
<i>nghb-sd</i>	0.000	0.149	0.446	0.143	0.251	0.454
<i>deep-im</i>	0.000	0.053	0.391	0.000	0.052	0.385
<i>mn2v-km</i>	0.000	0.084	0.424	0.000	0.080	0.415
<i>ts-net</i>	0.250	0.254	0.503	0.143	0.308	0.488

The results, averaged across the tested networks, are presented in Tab. 12 and Tab. 13. It is worth noting that the values for the *T* scenario are consistent across all metrics, as expected.

Notably, despite the different evaluation criteria employed, our method consistently achieves the top rank across all metrics. This performance aligns with the conclusions drawn in the main part of the paper and further underscores the robustness of the proposed model. At the same time, this evaluation methodology highlights the inherent difficulty of identifying super-spreaders in multilayer networks. The values obtained by all evaluated methods underscore both the complexity of the problem and the substantial room for further methodological advancements.

C Scalability of the model

To assess the scalability of the proposed model in terms of computational efficiency, two aspects were analysed: GPU memory consumption and processing time as a function of the network size. The evaluation was based on the test set, as it was also processed by the competing methods considered in this study.

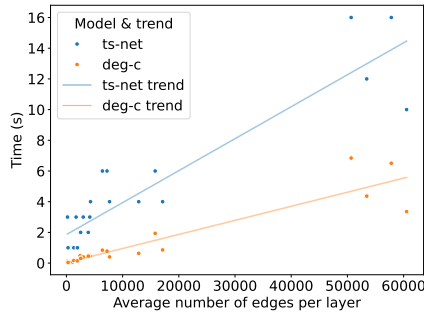


Figure 5: Inference time of *ts-net* compared to that of *deg-c*.

First, Fig. 5 presents processing times depending on the average number of edges per network layer for *ts-net* and, for reference, the light yet effective *deg-c*. The results for the proposed model indicate that processing time increases almost linearly, from near-zero values for small networks to approximately 16 seconds for networks containing over 60,000 edges per layer. In the case of *deg-c*, the scaling is also linear but with a gentler slope. Linear regression trends confirm these observations: *ts-net* exhibits a slope of 0.0002 with a strong fit ($R^2 = 0.853$), while *deg-c* shows a slightly lower slope of 0.00009 with $R^2 = 0.860$. It is worth noting that comparisons with *deg-c* on larger graphs would not be representative due to presumably lower-level optimisation mechanisms in the `degree` function

from the NetworkX library, which underpins this heuristic.

Regarding the second aspect, thanks to the use of `NeighborLoader`, the GPU memory usage of *ts-net* during inference remains constant at around 15.16 GB across the entire test set. This enables effective scaling and ensures applicability to real-world scenarios involving continuously growing networks.

D Training details

All *ts-net* variants were trained using the AdamW optimiser with a OneCycleLR learning rate scheduler. The implementation was developed in Python 3.12 using PyTorch 2.3.1. Training was limited to a maximum of 50 epochs, with early stopping applied and a patience threshold of 30 epochs. `NeighborLoader` with a batch size of 128, using an induced *N*-hop neighbourhood sampling strategy aligned with the number of convolutional layers was used. Specifically, up to 32 neighbours were sampled at the first hop, with the number halved at each subsequent hop. The maximum learning rate was set to 0.003. All experiments involving *ts-net* were conducted on a machine equipped with an Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz, 64 GB RAM, and an NVIDIA Titan RTX GPU. Hyperparameters for baseline machine learning methods were configured as described in their respective original papers and trained on CPU due to high memory requirements.

4.7 Multilayer Artificial Benchmark for Community Detection (mABCD)

Authors: Łukasz Kraiński, Michał Czuba, Piotr Bródka, Paweł Prałat, Bogumił Kamiński, François Théberge

Published at: Expert Systems with Applications; April 2026

MNiSW points: 200 (IF 7.5)

Citations: Web of Science: 0

(as of 6th March 2026) Scopus: 0

Google Scholar: 2

CRedit:

Methodology Software Validation Formal analysis
Investigation Data curation Writing – original draft
Writing – review & editing Visualisation



Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Multilayer artificial benchmark for community detection (mABCD)

Łukasz Kraiński^a, Michał Czuba^{b,*}, Piotr Bródka^b, Paweł Prałat^c,
Bogumił Kamiński^a, François Théberge^d

^a Decision Analysis and Support Unit, SGH Warsaw School of Economics, Warsaw, Poland^b Department of Artificial Intelligence, Wrocław University of Science and Technology, Wrocław, Poland^c Department of Mathematics, Toronto Metropolitan University, Toronto, ON, Canada^d Tutte Institute for Mathematics and Computing, Ottawa, ON, Canada

ARTICLE INFO

Keywords:

Synthetic graphs
Random graphs
Complex networks
Community structure
ABCD
Multilayer networks

ABSTRACT

One of the most persistent challenges in network science is the development of various synthetic graph models to support subsequent analyses. Among the most notable frameworks addressing this issue is the Artificial Benchmark for Community Detection (ABCD)—a random graph model with community structure and power-law distribution for both degrees and community sizes. The model generates graphs similar to the well-known LFR model but it is faster, more interpretable, and can be investigated analytically. In this paper, we use the underlying ingredients of ABCD and introduce its variant, mABCD, thereby addressing the gap in models capable of generating multilayer networks. The uniqueness of the proposed approach lies in its flexibility at both levels of modelling: the internal structure of individual layers and the inter-layer dependencies, which together make the network a coherent structure rather than a collection of loosely coupled graphs. In addition to the conceptual description of the framework, we provide a comprehensive analysis of its efficient Julia implementation. Finally, we illustrate the applicability of mABCD to one of the most prominent problems in the area of complex systems—spreading phenomena analysis.

1. Introduction

One of the most important features of real-world networks is their community structure, as it reveals the internal organization of nodes (Fortunato, 2010; Kamiński et al., 2021b). In social networks, communities may represent groups by interest; in citation networks, they correspond to related papers; in the Web graph, communities are formed by pages on related topics, etc. Identifying communities in a network is therefore valuable as this information helps us to understand the network structure better.

The need for synthetic graph models with community structure that resemble real-world networks arises for several reasons. Foremost, when modelling a given system is infeasible (for instance, due to data scarcity), such frameworks can provide a structure that globally approximates the analyzed environment. For example, in the field of epidemiology, the spread of diseases is commonly simulated on synthetic networks owing to the difficulty of obtaining ground-truth data at the human level (Wątroba & Bródka, 2023)—only recently have some countries developed network models resembling the social structures of their populations (Panayiotou et al., 2025). Continuing this example, simulat-

ing disease dissemination using the model proposed by Erdős et al. (1960) could be less accurate than doing so on a network generated by the stochastic block model (Holland et al., 1983), as the latter yields community-like structures that better reflect social bonds. Another motivation for developing synthetic graph generators with community structure lies in the need to benchmark and tune clustering algorithms that are unsupervised by nature, given that only a few datasets exist with ground-truth communities identified and labelled (Magnani et al., 2021a). Having such models enables experiments to be conducted in a controlled environment, thereby making the assessment of different community detection algorithms more reliable and reproducible. The generators proposed so far, although important milestones, have some limitations that can affect their applicability to different research scenarios (see Section 2.1 for details). These limitations underscore the ongoing need for new models, tailored to both more advanced approaches to modelling complex systems and the increasing size of networks used in analysis.

The LFR (Lancichinetti, Fortunato, Radicchi) model (Lancichinetti & Fortunato, 2009; Lancichinetti et al., 2008) is a highly popular model that generates networks with communities and, at the same time,

* Corresponding author.

E-mail addresses: lkrai@sgh.waw.pl (Ł. Kraiński), michal.czuba@pwr.edu.pl (M. Czuba), piotr.brodka@pwr.edu.pl (P. Bródka), pralat@torontomu.ca (P. Prałat), bkamins@sgh.waw.pl (B. Kamiński), theberge@ieee.org (F. Théberge).

<https://doi.org/10.1016/j.eswa.2025.130920>

Received 2 August 2025; Received in revised form 11 November 2025; Accepted 17 December 2025

Available online 19 December 2025

0957-4174/© 2025 Published by Elsevier Ltd.

allows for heterogeneity in the distributions of both node degrees and of community sizes. It became a standard and extensively used method for generating artificial networks. A similar synthetic network to LFR, the Artificial Benchmark for Community Detection (ABCD) (Kamiński et al., 2021a) was recently introduced and implemented,¹ including a fast implementation² that uses multiple threads (ABCDe) (Kamiński et al., 2022). Undirected variants of LFR and ABCD produce graphs with comparable properties, but ABCD/ABCDe is faster than LFR and can be easily tuned to allow the user to make a smooth transition between the two extremes: pure (disjoint) communities and random graphs with no community structure. Moreover, it is easier to analyze theoretically—for example, in Kamiński et al. (2022) various theoretical asymptotic properties of the ABCD model are investigated, including the modularity function that, despite some known issues such as the “resolution limit” reported in Fortunato and Barthelemy (2007), is an important graph property of networks in the context of community detection. In (Barrett et al., 2025b), some interesting and desired self-similar behaviour of the ABCD model is analyzed; namely, it is shown that the degree distribution of ground-truth communities is asymptotically the same as the degree distribution of the whole graph (appropriately normalized based on their sizes). Finally, the building blocks in the model are flexible and may be adjusted to satisfy different needs. Indeed, the original ABCD model was recently adjusted to include potential outliers (ABCD + o) (Kamiński et al., 2023a), overlapping communities (ABCD + o²) (Barrett et al., 2025a), and extended to hypergraphs (h-ABCD) (Kamiński et al., 2023b).³ The ABCD model is used by practitioners but, for the reasons mentioned above, it also gains recognition among scientists. For example, Aref et al. (2022) suggests to use Adjusted Mutual Information (AMI) between the partitions returned by various algorithms with the ground-truth partitions of synthetically generated random graphs, ABCD and LFR. In particular, they use both models to compare 30 community detection algorithms, mentioning that *being directly comparable to LFR, ABCD offers additional benefits, including higher scalability and better control for adjusting an analogous mixing parameter.* In the context of this paper, the most important of the above features is that the ABCD model is flexible and can be easily extended, as we utilize that property to deliver a new model to producing multilayer networks.

The study of complex networks has evolved significantly over the past two decades, driven by the growing need to model and analyze interconnected systems. Among the notable advancements in this field is the development of multilayer networks, a powerful framework that captures multiple types of relationships within a single network structure. Unlike traditional single-layer (a.k.a. monoplex) networks where all edges represent the same type of interaction, multilayer networks allow nodes to participate in diverse types of relationships across different layers. For instance, in a multilayer social network, layers may represent distinct relations such as friendships, coworker connections, family ties, and online interactions via platforms like Facebook or LinkedIn (Dickison et al., 2016; Kivelä et al., 2014). This flexibility makes multilayer networks uniquely suited to model the complexity of real-world systems. The concept of multilayer networks was introduced to address the limitations of traditional network models, particularly their inability to represent systems with heterogeneous interactions. Early research focused on formalizing the structure of multilayer networks, including their mathematical representation and classification. Kivelä et al. (2014) provided a seminal framework for multilayer networks, categorizing them into multiplex, interconnected, and temporal networks. This foundational work established the groundwork for studying the dynamics and structures of multilayer systems, sparking a surge of interest in the network science community.

As we already mentioned, one of the most important aspects of network analysis is community detection, which aims to identify groups of nodes that are densely connected within the network while being sparsely connected to nodes outside their group (De Luca et al., 2023). In multilayer networks, this problem is significantly more complex because communities may exist consistently across layers, vary between layers, or even exhibit dependencies across layers (Magnani et al., 2021a). The field of community detection in multilayer networks emerged shortly after the formalization of multilayer network structures, with pioneering studies exploring both theoretical and algorithmic approaches (Berlingerio et al., 2011; Bródka et al., 2013; Tang et al., 2012). The survey by Magnani et al. (2021a) provides an exhaustive overview of existing methods for community detection in multilayer networks, categorizing them into optimization-based methods (e.g., modularity maximization), statistical methods (e.g., stochastic block models), and heuristic approaches. These methods address diverse challenges, such as detecting overlapping communities, identifying temporal changes, and incorporating inter-layer dependencies. Despite these advancements, the problem of community retrieval in multilayer networks remains pertinent, and new approaches continue to emerge (Karimi et al., 2020; Qing & Wang, 2024). One of the major challenges associated with this task is the evaluation of algorithms, primarily due to the scarcity of real-world multilayer datasets with ground-truth communities. In the aforementioned review study, Magnani et al. (2021b) recalled only two multilayer networks with partially known partitions, which, however, do not arise from their geometry but from actor labels assigned according to affiliation (e.g., political party). Moreover, both networks were constructed such that their communities are similar (i.e., pillar-like) across layers, which makes them insufficiently representative for benchmarking diverse community detection algorithms.

In this paper, we use the underlying flexible ingredients of the ABCD family of models and introduce its variant for multilayer networks, **mABCD**^{4, 5}. It is a journal version of a short conceptual paper (Bródka et al., 2025) presented at the 20th Workshop on Modelling and Mining Networks (WAW2025). We significantly extend the previous work by improving, evaluating, and validating the model as well as setting **mABCD** in the context of related works. In summary, the main contributions of this work can be expressed in five key points:

- The proposed model, **mABCD**, is capable of generating synthetic, scale-free multilayer networks with community structure, where structural properties are shaped both within individual layers and across them.
- We introduce a latent, biscuit-like layer that supports the process of node assignment into communities and is shared across all layers during the generation process. This structure represents intrinsic actor properties and can be readily replaced with embeddings of real-world entities, thereby enhancing the model's flexibility.
- The proposed model extends and strengthens the broader ABCD family of synthetic graph generators, contributing to its coherence and generality.
- **mABCD** is an efficient model that outperforms its closest equivalent, multilayerGM, by one order of magnitude in time needed to generate small networks (around 1000 nodes), with the improvement increasing up to two orders of magnitude for larger networks (around 33,000 nodes).
- We demonstrate a potential application of the model by addressing the problem of spread controllability in multilayer networks.

The paper is structured as follows. We start with a gentle introduction to various aspects of multilayer networks (Section 2), including an

¹ <https://github.com/bkamins/ABCDGraphGenerator.jl/>

² <https://github.com/tolcz/ABCDeGraphGenerator.jl/>

³ <https://github.com/bkamins/ABCDHypergraphGenerator.jl>

⁴ <https://github.com/Krainiskil/MLNABCDGraphGenerator.jl>

⁵ Please note that the name of the proposed method is intended to emphasize its connection to the ABCD model rather than to draw the reader's attention to benchmarking different community detection methods.

overview of existing synthetic models, summarizing approaches to measure correlations between layers. Then, we present design assumptions under which **mABCD** was developed (Section 3). In Section 4, we formally define the **mABCD** model. Experiments highlighting properties of the model are discussed in Section 5. We finish the paper with a demonstration of how one can use the **mABCD** model to investigate properties of complex networks and algorithms that are running on them. As an example, we concentrate on information spread (Section 6), which constitutes a problem of greater complexity in the realm of multilayer networks compared to its monoplex counterpart.

2. Related work

In this section, we introduce the standard notation used across the paper and properties of interest, especially from the perspective of multilayer networks (Dickison et al., 2016; Kivelä et al., 2014). For a given $n \in \mathbb{N} = \{1, 2, \dots\}$, we use $[n]$ to denote the set consisting of the first n natural numbers, that is, $[n] = \{1, 2, \dots, n\}$. We define a multilayer network as a quadruple $M = ([n], [\ell], V = [n] \times [\ell], E)$, where:

- $[n]$ is a set of n actors (for example, users of various social networking sites),
- $[\ell]$ is a set of layers (for example, different social networking platforms, such as LinkedIn, Facebook and Instagram, on which actors interact with each other),
- $V \subseteq [n] \times [\ell]$ a set of nodes (vertices); node $v = (a, \ell_i) \in V$ represents an actor a in layer ℓ_i ,
- E is a set of (undirected) edges between nodes; if $e = v_1 v_2 \in E$ with $v_1 = (a_1, \ell_1) \in V$ and $v_2 = (a_2, \ell_2) \in V$, then $\ell_1 = \ell_2$, that is, edges occur only within layers.

Fig. 1 presents an example of a simple multilayer network consisting of eleven actors (represented as integers from [11]), three layers (professional, associated with the actors' interactions in a work environment, friendship relations between actors, and a layer representing actors playing football together), thirty nodes, and thirty edges.

Note that not every actor in the example above is present on all layers. For example, actor 8 does not exist in the football layer, meaning that it does not play football with other actors. For simplicity, in our model, we assume that each layer has exactly n nodes associated with all actors. Actors that do not engage with a given layer (we will call them inactive) will be associated with isolated nodes (nodes of degree zero).

2.1. Existing synthetic models for multilayer networks

The first synthetic models for multilayer networks, with known community structure, were extensions of existing single-layer benchmarks. For example, in Bródka (2016), the **mLFR** model was introduced, as an extension of the original **LFR** benchmark, to generate multilayer networks with realistic (scale-free) degree and community size distributions. While the **mLFR** model preserves realistic degree and community size distributions, it assumes homogeneity in community generation across layers, limiting its applicability to networks where community structures vary between layers. Another drawback is its limited ability to model inter-layer correlations, a critical aspect of many real-world multilayer networks. Furthermore, the **mLFR** benchmark inherits the computational complexity of the original **LFR** model, making it resource-intensive for generating large networks with detailed configurations. Another example of early works presenting multilayer network generators is (Popiel et al., 2015), the **MuNeG** generator that was proposed as an extension of the Eldardiry model (Eldardiry & Neville, 2010). It is governed by six scalar parameters: the number of nodes, layers, and groups; the probability that two nodes from the same (or different) group are connected; and community homophily. However, as an early contribution to the field of multilayer network generation,

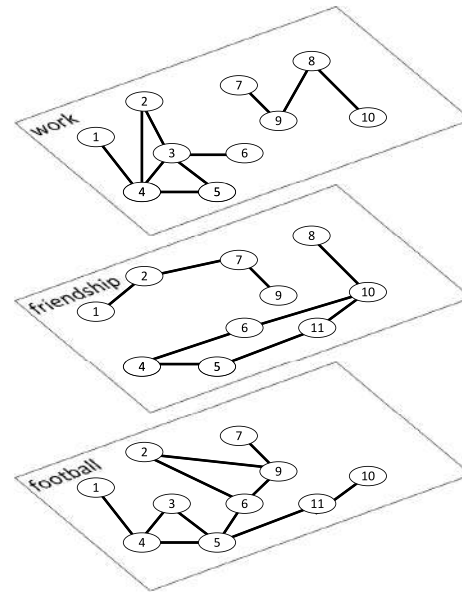


Fig. 1. Example of a multilayer network.

its functionality is limited. The model does not capture inter-layer relations, nor does it provide direct control over the distributions of degrees or communities.

As the body of literature on multilayer networks expanded, various generators addressing specific forms of multilayer networks emerged. For example, Tarvid (2017) introduces a synthetic model designed to construct multilayer networks in which a link between two actors may exist in only one layer, thereby representing a social system composed of four distinct types of relationship: support, sympathy, band, and community. Elsisy et al. (2022) also proposed a generator tailored to a specific problem. Specifically, they developed a framework for modelling covert social structures, such as criminal or terrorist networks. Their goal was to provide a model capable to represent both the organizational structure and the community partition of the considered system. These models, despite useful for specific problems, remain limited in terms of their applicability to more general scenarios.

A review of related work also reveals a broad influx of methods extending the Stochastic Block Model (Holland et al., 1983). For instance, Huang et al. (2019) build upon this framework by assuming that a given community may appear in multiple layers, and propose a method for retrieving such exactly matching groups from the network. On the other hand, the generator proposed by Pamfil et al. (2019) emerges as an indirect outcome of efforts to generalize the modularity function to a multilayer setting. It attempts to address several limitations of extensions of the Stochastic Block Model, such as the restriction to identical communities across layers, but imposes an additional constraint: any inter-layer dependencies are induced solely through the multilayer partition and the coupling of nodes. Another significant contribution (also based on the Stochastic Block Model) came from Bazzi et al. (2020), who proposed a framework—**multilayerGM**—for generating networks with mesoscale structures, including communities. This is a very general and powerful model that enables researchers to control the properties of generated networks, such as the strength of inter-layer connections and the overlap of communities between layers. Because of its versatility (multilayer, temporal networks and more), the model might be challenging to use for a novice researcher (the need to define an inter-layer dependency tensor, implementing a new network model if one wants to use something else than the Degree Corrected Stochastic Block Model) and

it suffers from high computational complexity (see Fig. 17 and Table 4). Finally, it lacks the flexibility needed to reflect the various dependencies existing in real networks (for example, injecting degree sequences from real multilayer networks). As a result, despite its advanced capabilities, the model does not address all needs.

Magnani et al. (2021a), in the recent review paper on community detection in multiplex networks, have not used the model proposed by Bazzi. Instead, they proposed a simple model for generating multilayer networks with community structures designed specifically for various algorithm comparisons. This model, partially integrated into the multinet library (Magnani et al., 2021b), simplifies the generation of multilayer networks and facilitates the benchmarking of various community detection algorithms. While this simplicity facilitates comparisons of different algorithms, it limits the model's ability to replicate the complexity of real-world networks, such as diversity in community sizes. Furthermore, the model offers only basic representations of inter-layer dependencies, which reduces its relevance for studying networks with varying edge correlations between layers.

3. Design assumptions

Before building the model, it is essential to address the question if and how degrees, edges, and partitions correlate between layers in real-world networks. Existing studies (Bródka et al., 2018; Magnani et al., 2021a) have not provided a definitive answer to this question. To bridge this gap, we conducted an analysis of eight real-world networks from diverse domains and of different sizes. In the following section, we first present a methodology for computing these correlations. Then, we discuss obtained results and conclude with the assumptions underlying the design of **mABCD**, which stem from this analysis.

3.1. Correlations between layers

Although there are no edges between nodes in different layers, in most real-world multilayered networks, layers are clearly *not* independently generated. Each actor is associated with ℓ nodes, one in each layer, and there are some highly non-trivial correlations between edges across layers. For example, active users on one social media platform are often also active on another one (Gottfried, 2024). This creates correlations between degree distributions across layers. Communities that are naturally formed in various layers often depend on the properties of the associated actors. For example, users interested in soccer might group together on Instagram and on Facebook. As a result, partitions of nodes into communities (associated with different layers) are often correlated. Finally, interactions between actors in one layer might increase their chances of interacting in another layer, yielding correlations at the level of edges.

Below, we summarize how we measure these three types of correlations. The first two measures are standard, and their description can be found in Bródka et al. (2018), Kamiński et al. (2021b).

3.1.1. Correlations between node degrees in various layers

We will use **Kendall rank correlation coefficient** τ (Kendall, 1938) to measure correlations between sequences of node degrees in two different layers. It is a nonparametric measure of the ordinal association between two measured quantities: the similarity of the orderings of the data when ranked by each of the quantities (in our application, the degree sequences). The Kendall correlation between two variables ranges from -1 to 1 . It is large when observations have a similar rank between the two variables and is small when observations have a dissimilar rank. Specifically, we will use the “tau-b” statistic, which is adjusted to handle ties. If an actor is inactive in one of the two layers we compare against each other, then we simply ignore the two nodes corresponding to this actor. As a result, the degree sequences are always of the same length.

3.1.2. Correlations between partitions in various layers

The **adjusted mutual information** (AMI), a variation of **mutual information** (MI), is a common way to compare partitions of the same set (Kamiński et al., 2021b; Vinh et al., 2009). Usually, one may want to compare the partitions returned by some clustering algorithms. In our present context, we may want to compare partitions into ground-truth communities from two different layers. The AMI takes a value of 1 when the two partitions are identical and 0 when the MI between two partitions equals the value expected due to chance alone. Actors, that are inactive in at least one of the two layers we compare against each other, are ignored so that a comparison of partitions is made on the same set of actors.

3.1.3. Correlations between edges in various layers

To measure correlations between edges in different layers, we define $\mathbf{R}$, a $\ell \times \ell$ matrix in which elements $r_{i,j} \in [0, 1]$ ($i, j \in [\ell]$) capture correlation between edges present in layers i and j . For any $i, j \in [\ell]$ with $i < j$, let

$$E_i^j = \{a_1 a_2 : (a_1, i)(a_2, i) \in E \wedge a_1, a_2 \in [n] \wedge a_1 \text{ and } a_2 \text{ are active in layers } i \text{ and } j\}, \quad (1)$$

be the set of edges that are present in layer i , involving actors that are also active in layer j . Note that in the definition of E_i^j , edges are defined over actors that are active in both layers, not nodes in layer i , so that we can perform set operations on edges between layers. Entries $r_{i,j}$ in $\mathbf{R}$ are computed using the following formula:

$$r_{i,j} = \frac{|E_i^j \cap E_j^i|}{\min\{|E_i^j|, |E_j^i|\}}. \quad (2)$$

If $\min\{|E_i^j|, |E_j^i|\} = 0$, then we leave $r_{i,j}$ undefined; in the implementation, NaN value is produced.

Note that the definition of $\mathbf{R}$ implies that $r_{i,i} = 1$ for any $i \in [\ell]$ and $r_{i,j} = r_{j,i}$ for $1 \leq i < j \leq \ell$. The maximum value of 1 is attained when edges in one of the layers form a subset of edges in the other layer. The minimum value of 0 is attained when the two sets of edges in the corresponding layers are completely disjoint. As a result, r_{ij} aims to capture correlations between individual edges, but it is not normalized as, for example, the Kendall rank correlation coefficient τ . The coefficient τ ranges from -1 to 1 , corresponding to the two extremes, and 0 corresponds to a neutral case. Graphs associated with the layers are sparse, but one layer might have substantially more edges than the other. Hence, r_{ij} is convenient, but it does not have a natural interpretation as τ . Finally, let us mention that one can easily update the value of r_{ij} when some small operations are applied to either layer i or j . It will become handy when such operations must be performed on our synthetic model to converge to the desired correlation matrix $\mathbf{R}$.

3.2. Examples of multilayer networks

The aspects of inter-layer dependencies, in the form introduced above, were derived from eight real-world networks, which varied in the number of actors, nodes, edges, layers, and in the domains they represented. Table 1 presents a detailed overview of the characteristics of each network, offering insights into their structural properties.

Next, in Fig. 2, we present correlation matrices between layers for degrees of nodes, partitions, and edges, according to the methodology described in Section 3.1. Due to a lack of ground truth partitions, communities were identified by the outcome of the Louvain algorithm (Blondel et al., 2008). After conducting a thorough analysis of the results, we observed a lack of consistent patterns in the examined real-world multilayer networks. Specifically, there was no universal relationship between degrees, edges, and community partitions across the layers. In

Table 1

Real-world networks evaluated with respect to their inter-layer dependencies, with their basic parameters summarized.

Name	Layers	Actors	Nodes	Edges	Note
arxiv	13	14,065	26,796	59,026	Coauthorship network obtained from articles published on the “arXiv” repository (De Domenico et al., 2015). Each layer represents a different arXiv category, e.g. Physics and Society or Social and Information Networks.
aucs	5	61	224	620	A graph of interactions (friends on Facebook, leisure, work, co-authorship and lunch) between employees of Aarhus University, Department of Computer Science (Rossi & Magnani, 2015).
cannes	3	438,537	659,951	974,743	A network of interactions (retweets, mentions and replies) between Twitter (now X) users during the Cannes Film Festival in 2013 (Omodei et al., 2015).
ckmp	3	241	674	1370	A network depicting diffusion of innovations among physicians (Coleman et al., 1957). Each layer was built based on the physician’s answers to questions like “Tell me the first names of your three friends whom you see most often socially”.
eutr-A	37	417	2034	3588	The European air transportation network (Cardillo et al., 2013). Each layer represents the connections of different airline operator in Europe.
l2-course	2	41	82	297	A network of interactions (collaboration, friends) between U.S. students learning Arabic language during an intensive course in Jordan (1st month snapshot) (Paradowski et al., 2024).
lazega	3	71	212	1659	A network of interactions (co-work, friendship and advice) between staff of a law corporation (Snijders et al., 2006).
timik	3	61,702	102,247	881,676	A graph of interactions (text messages, online transactions, home visits) between users of the virtual world platform for teenagers (Jankowski et al., 2017).

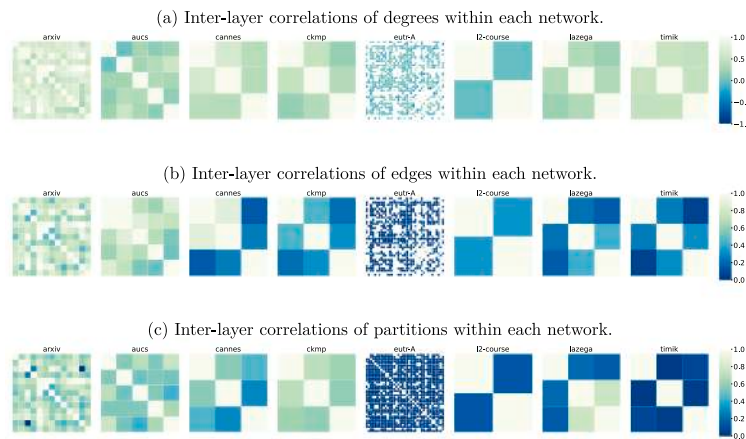


Fig. 2. Correlations between degrees, edges, and partitions presented as heatmaps for real-world networks: arxiv, aucs, cannes, ckmp, eutr-A, l2-course, lazega, and timik (layer names and explicit values have been removed for clarity). Note that there is no consistent trend across networks or within the given property.

some cases, these features were correlated, while in others, they appeared to be independent. Furthermore, within a single network, we found that while two layers might exhibit a correlation, this relationship did not necessarily extend to other layers. For instance, a node might have similar neighbours across two layers but exhibit a completely different connectivity pattern in a third layer. These observations highlight the inherent heterogeneity and complexity of real-world multilayer networks, emphasizing the need for a flexible and customizable synthetic network generation framework.

3.3. Assumed functionalities

In light of these findings, we assumed that the framework has to accommodate inter-layer diversities. Moreover, since the question of how precisely these relations should be modelled constitutes an issue that must be addressed individually, depending on the specific problem one is working on, we aimed to provide a tool that balances flexibility and accessibility. Thus, building the framework upon ABCD seemed a plausible approach, as it allows for extension in this direction.

On the one hand, **mABCD** aims to provide advanced users with full control over critical properties of the network, such as the distributions of degrees or edges correlations. This level of customization ensures that users can generate networks tailored to their specific research needs, allowing for precise modelling of complex scenarios observed in real-world systems. On the other hand, we recognized the importance of making the framework accessible to less experienced users who may not be familiar with the intricacies of network science and programming. To address this, **mABCD** was designed to include default settings, like generating networks with a power-law degree distribution, which is a common feature of many real-world systems. These default parameters provide a starting point for users while maintaining the ability to modify them as needed. Additionally, this is complemented by guidelines, default configurations, and examples provided at the associated GitHub repository.

In conclusion, by combining flexibility with ease of use, we designed **mABCD** to bridge the gap between accessibility and sophistication, making it a versatile tool for researchers across different levels of expertise. This approach ensures that users can generate synthetic multilayer networks that reflect the diverse and sometimes unpredictable patterns observed in real-world systems while still being able to explore a wide range of scenarios, from highly correlated layers to those with minimal interdependence. That kind of adaptability constitutes a crucial step forward in advancing the study and validation of community detection algorithms and other analytical techniques in multilayer network science.

4. The mABCD model

In this section, we introduce a variant of the **ABCD** model that produces a synthetic collection of graphs that form a multilayer structure, **mABCD**. The subsequent text guides through the details of the network construction process. Its parameters are summarized in Tables 2 and 3, while the process itself is conceptually illustrated in Fig. 3.

4.1. Preliminaries

Since the networks produced by the proposed framework follow a power-law distribution in both node degrees and community sizes, and since their generation proceeds sequentially, we first introduce the preliminary concepts that are fundamental to the construction process.

4.1.1. Power-law distribution

Power-law distributions will be used to generate both the degree sequence and community sizes. For given parameters $\gamma \in (0, \infty)$, $\delta, \Delta \in \mathbb{N}$ with $\delta \leq \Delta$, we define a truncated power-law distribution $\mathcal{P}(\gamma, \delta, \Delta)$ as follows. For $X \sim \mathcal{P}(\gamma, \delta, \Delta)$ and for $k \in \mathbb{N}$ with $\delta \leq k \leq \Delta$,

$$\mathbb{P}(X = k) = \frac{\int_k^{k+1} x^{-\gamma} dx}{\int_\delta^{\Delta+1} x^{-\gamma} dx}.$$

4.1.2. The configuration model

Suppose then that our goal is to create a graph on n nodes with a given degree sequence $\mathbf{d} = (d_i, i \in [n])$, where $\mathbf{d}$ is a sequence of non-negative integers such that $m = \sum_{i \in [n]} d_i$ is even. We define a random multi-graph $\text{CM}(\mathbf{d})$ with a given degree sequence known as the **configuration model** (sometimes called the **pairing model**), which was first introduced by Bollobás (1980). See: Bender and Canfield (1978), Wormald (1984), Wormald et al. (1999) for related models and results.

We start by labelling nodes as $[n]$ and, for each $i \in [n]$, endowing node i with d_i half-edges. We then iteratively choose two unpaired half-edges uniformly at random (from the set of pairs of remaining half-edges) and pair them together to form an edge. We iterate until all half-edges have been paired. This process yields $G_n \sim \text{CM}(\mathbf{d})$, where G_n is allowed self-loops and multi-edges and thus G_n is a multi-graph.

Table 2

Global parameters of **mABCD**.

Parameter	Range	Description
n	$\mathbb{N}$	Number of actors (recommended $n \geq 1000$)
ℓ	$\mathbb{N}$	Number of layers
$\mathbf{R}$	$[0, 1]^{\ell \times \ell}$	Inter-layer edge correlation matrix
d	$\mathbb{N}$	Dimension of reference layer

4.2. Parameters of the mABCD model

The **mABCD** model is governed by the following parameters. The first family of parameters is responsible for a few global properties of the model and is listed in Table 2.

Actors will be associated with *labels* from the set $[n]$. These labels will affect the degrees of actors. Each actor $a \in [n]$ will be associated with ℓ nodes, $v_i = (a, i)$ with $i \in [\ell]$, one for each of the ℓ layers. Moreover, each actor will be associated with a vector in $\mathbb{R}^d$ representing their features. We will refer to these vectors as vectors in the *reference layer*. This reference layer will affect the process of generating partitions into communities in various layers.

The second family of parameters (Table 3) is responsible for various properties that are specific to each of the ℓ layers; subscripts $i \in [\ell]$ indicate that the corresponding parameters shape the i^{th} layer. In particular, the set of parameters ξ_i , $i \in [\ell]$, will control the level of noise, that is, the fraction of edges in layer i that are between nodes from two different communities. The suggested range of values for parameters γ and β (namely, the intervals $(2, 3)$ and $(1, 2)$, respectively) are chosen according to experimental values commonly observed in complex networks (Barabási, 2016; Orman & Labatut, 2009).

4.3. The mABCD construction

We will use $\mathcal{A}$ for the distribution of graphs (layers) generated by the following 6-phase construction process. The model generates ℓ graphs; graph $G_n^i = ([n] \times \{i\}, E^i)$, $i \in [\ell]$, is the graph representing the i^{th} layer. Once they are generated, we simply take $V = \bigcup_{i \in [\ell]} ([n] \times \{i\})$ and $E = \bigcup_{i \in [\ell]} E^i$.

Conceptually, the phases of network construction are illustrated in Fig. 3. The first five steps are carried out independently for each of the ℓ layers, while the sixth binds the generated graphs into a single multilayer network. The process is governed by the parameters listed in Tables 2 and 3. In the first step, the so-called active nodes (i.e., those that will not be isolated) are determined. Each active node is then assigned its degree value. During the third step, communities are created and populated with nodes according to the latent, biscuit-like layer. Phase four focuses on connecting nodes within the same community as well as establishing inter-community (i.e. background) links. These graphs are subsequently simplified to remove self-loops and multiedges while preserving the intended structural properties. The final phase performs a series of edge rewirings to achieve the targeted edge correlations between each pair of layers. The illustration is complemented by pseudocode published on GitHub repository.

4.3.1. Phase 1: selecting active nodes

As mentioned above, in a multilayer network, not all of the actors are active in all the layers. Actor $a \in [n]$ is *active* in layer i with probability q_i , independently for each $i \in [\ell]$ and all other actors. If an actor is not active in a given layer, then it will be represented by an “artificial” *inactive* node. We use N^i to denote the number of active nodes (and actors) in layer i . (Clearly, N^i is a random variable with expectation $q_i n$.) For convenience, we will keep inactive nodes as a part of the corresponding graphs, but one may think of them as being removed from a given layer.

4.3.2. Phase 2: creating degree sequences

The degree sequences for all of the ℓ layers are generated independently so we may concentrate on a given layer $i \in [\ell]$. We ensure that

Table 3
Layer-specific parameters of **mABCD** (the index denotes the i th layer).

Parameter	Range	Description
q_i	$(0, 1]$	Fraction of active actors
τ_i	$[-1, 1]$	Correlation coefficient between degrees and labels
r_i	$[0, 1]$	Correlation strength between communities and the reference layer
γ_i	$(2, 3)$	Power-law degree distr. with exponent γ_i
δ_i	$\mathbb{N}$	Min. degree at least δ_i (recommended $\delta_i \geq 10$)
Δ_i	$\mathbb{N} (1 \leq \delta_i \leq \Delta_i < n)$	Max. degree at most Δ_i
β_i	$(1, 2)$	Power-law community size distr. with exponent β_i
s_i	$\mathbb{N}$	Min community size at least s_i (recommended $s_i \geq 50$)
S_i	$\mathbb{N} (\delta < s_i \leq S_i \leq n)$	Max community size at most S_i
ξ_i	$(0, 1)$	Level of noise

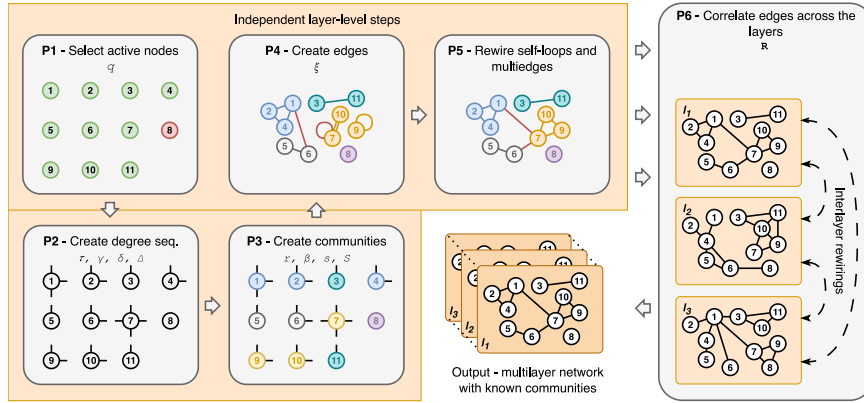


Fig. 3. A schematic illustration of the **mABCD** model's operating principle, exemplified by the generation of a three-layer network. The process consists of six steps. The first five are performed independently for each layer, yielding scale-free monoplex graphs with a known community structure drawn from a shared latent space. In the final step, the layers are glued through a sequence of rewirings to achieve the desired cross-layer edge correlations.

the degree sequence satisfies (a) a power-law with parameter γ_i , (b) a minimum value of at least δ_i , and (c) a maximum value of at most Δ_i (see Section 4.1).

Inactive nodes (representing actors not present in particular layer) are easy to deal with, they simply have degree zero. The remaining N^i degrees are i.i.d. samples from the distribution $\mathcal{P}(\gamma_i, \delta_i, \Delta_i)$. We use $\mathbf{d}_n^i = (d_v^i, v \in [n])$ for the generated degree sequence of G_n^i with $d_1^i \geq d_2^i \geq \dots \geq d_n^i$; $\mathbf{d}_{N_i}^i$ is a degree subsequence of active nodes. Finally, to ensure that $\sum_{v \in [n]} d_v^i$ is even, we decrease d_1^i by 1 if necessary; we relabel as needed to ensure that $d_1^i \geq d_2^i \geq \dots \geq d_n^i$.

Parameter $\tau_i \in [-1, 1]$ controls how degrees of the nodes are correlated with labels of the associated actors (recall that node (a, i) in layer i is associated with an actor with label a). In one important case, namely, when $\tau_i = 0$, there is no correlation at all and the degree sequence $\mathbf{d}_{N_i}^i$ is assigned randomly to the N_i active nodes. When $\tau_i = 1$, the order of active nodes with respect to their labels is the same as the order with respect to their degrees; the largest degree node is first. In other words, if nodes $(a_1, i), (a_2, i)$ with $1 \leq a_1 < a_2 \leq n$ are active, then $\deg^i(a_1) \geq \deg^i(a_2)$, where $\deg^i(a_1)$ is the degree of node (a_1, i) . Similarly, if $\tau_i = -1$, then the order of active nodes with respect to their labels is also consistent with the order with respect to their degrees but this time the last node is of the largest degree. Since τ_i 's could be different for different layers, one node could have large degrees in some layers but small ones in some other ones.

To achieve the desired property, each active node (a, i) independently generates a normally distributed random variable $X_a = N(a/n, \sigma_i)$, where the variance σ_i is a specific function of τ_i . (Recall that we concentrate on a given layer $i \in [\ell]$. For convenience, we simplify the notation and stop referencing to layer i in notation such as X_a . Still, there are many independent random variables for each active node (a, i)

associated with actor a .) We sort active nodes in increasing order of their values of X_a and assign the degree sequence accordingly; that is, node (a, i) gets degree d_r^i , where $r \in [N^i]$ is the rank of X_a . In particular, the node with the smallest value of X_a gets assigned the largest degree, namely, d_1^i . Note that if $\sigma_i = 0$, then $X_a = a/n$ (deterministically), and so we recover the perfect correlation between the degrees and the labels ($\tau_i = 1$). On the other hand, if $\sigma_i \rightarrow \infty$, then the order of nodes is perfectly random (with uniform distribution), so we recover the other desired extreme ($\tau_i = 0$).

Function $\sigma_i : [0, 1] \rightarrow [0, \infty)$ is empirically approximated so that the variance $\sigma_i = \sigma_i(\rho_i)$ yields the Kendall rank correlation close to $\tau_i \in [0, 1]$ between the ordering generated by the ranks of X_a and the labels a associated with corresponding actors that are active in layer i . Twenty degree distributions are independently generated (with the same σ and different random seeds), and the one with the correlation coefficient that is the closest to the desired value of τ_i is kept. To deal with negative correlations $\tau_i \in [-1, 0)$, we simply “flip” the order generated for $|\tau_i|$.

4.3.3. Phase 3: creating communities

Our next goal is to create community structure in each layer of the **mABCD** model. When we construct a community, we assign a number of nodes to said community equal to its size. Initially, the communities form empty graphs. Then, in later phases we handle the construction of edges using the degree sequence established in Phase 2.

Similarly to the process of generating the degree sequences, the sequence of community sizes are generated independently, ensuring that the distribution for a given layer $i \in [\ell]$, satisfy (a) a power-law with parameter β_i , (b) a minimum value of s_i , and (c) a maximum value of S_i (see Section 4.1). In addition, we also require that the sum of community sizes is exactly n . Specifically, inactive nodes (if there are any) form

their own community, namely, C_0^i . Other communities are generated with sizes determined independently by the distribution $\mathcal{P}(\beta_i, s_i, S_i)$. We generate communities until their collective size is at least n . If the sum of community sizes at this moment is $n + x$ with $x > 0$, then we perform one of two actions. If the last added community has a size at least $x + s_i$, then we reduce its size by x . Otherwise (that is, if its size is $c < x + s_i$), we delete this community, select $c - x$ old communities and increase their sizes by 1.

Now, given that the sequences of community sizes are already determined (for all layers), it is time to assign nodes to communities. To allow communities to be correlated with each other, we first create a latent reference layer that will guide the process of assigning nodes to specific communities across all layers. One may think of this auxiliary layer as properties of actors (such as people's age, education, geographic location, beliefs, etc.) shaping different layers (for example, various social media platforms). This single reference layer will be used for all ℓ layers. In this reference layer, each actor $a \in [n]$ gets assigned a random vector in $\mathbb{R}^d$ (by default, $d = 2$) that is taken independently and uniformly at random from the ball of radius one centred at $\mathbf{0} = (0, 0, \dots, 0)$.

Let us now concentrate on a given layer $i \in [\ell]$. Recall that the community sizes have already been generated. We write L^i for the (random) number of regular communities in layer i partitioning the set of active nodes in this layer and use $c^i = (c_j^i, j \in \{0\} \cup [L^i])$ for the corresponding sequence of community sizes. Recall that inactive nodes (representing actors not present in a particular layer) form their own community (namely, C_0^i) so $c_0^i = n - N^i$ is the number of inactive nodes in layer i . Let R be the set of active nodes. We assign nodes to communities, dealing with one community at a time, in a random order. When community C_j^i is formed (for some $j \in [L^i]$), we first select a node from R that is at the largest distance from the center $\mathbf{0}$ (in the reference layer). This node, together with its $c_j^i - 1$ nearest neighbours in R , are put to C_j^i . We remove C_j^i from R and move on to the next community.

The above strategy creates a partition of nodes that is highly correlated with the geometric locations of nodes in the reference layer; nodes that are close to each other in the reference layer are often in the same community. An example of this process is illustrated in Fig. 4, which shows how nodes are assigned to communities in two network layers, both guided by the same reference layer. This mechanism can be intuitively compared to eating a biscuit, bite by bite, from the edge towards the centre. Nodes that are close to each other are more likely to be "eaten at the same time" and thus placed in the same group. Moreover, the severed pieces may vary in size, so do communities. Finally, the same biscuit can be enjoyed by two different gourmets, which in this analogy corresponds to reusing the same reference layer for each of the relationships modelled within the generated network. To reduce the correlation strength (modelled by the parameter $r_i \in [0, 1]$), we perform the following procedure. Each active node independently leaves its own

community with probability $1 - r_i$, freeing a spot in this community. All the nodes that left are then put back randomly to any available spot (which typically is in a community that this node was *not* originally in). Note that in the extreme case, when $r_i = 0$, the resulting partition does not depend on the reference layer at all, so there is no correlation. We write $\mathbf{C}_n^i = (C_j^i, j \in \{0\} \cup [L^i])$ for the generated collection of communities in G_n^i . Again, let us stress the fact that $\mathbf{C}_n^i$ is a random partition of $[n]$ of random size $L^i + 1$. At this point, it is worth mentioning that, due to the shared latent layer and a parameter r_i specific to each relationship, it is possible to model the alignment of communities across the network. Following (Magnani et al., 2021a), where the notion of a *pillar* is introduced to describe a community that persists across at least two layers, layers with $r_i \approx 1$ tend to exhibit this form of cluster alignment. Conversely, if the values of r_i are set to more distinctive numbers, the communities are expected to be largely disjoint across the layers.

Finally, note that in the above process of assigning nodes to communities, as opposed to the original ABCD model, we ignore the degree of nodes. Indeed, the original ABCD model tries to make sure that large degree nodes are not assigned to small communities. In **mABCD**, there are many layers and non-trivial correlations between partitions into communities and degree sequences between layers. In a hypothetical extreme situation, it might happen that each node belongs to some small community in some layer. Hence, the **mABCD** model does not try to prevent such unavoidable situations and will resolve potential issues later (see Phase 5).

4.3.4. Phase 4: creating edges

Now, it is time to form edges in **mABCD**. It will be done in the next three phases. Phases 4 and 5 will independently generate ℓ graphs G_n^i , $i \in [\ell]$, for each of the ℓ layers whereas Phase 6 will make sure that edges across various layers are correlated, if needed. We may then concentrate on a given layer $i \in [\ell]$.

At this point G_n^i contains n nodes labelled as (a, i) , $a \in [n]$, partitioned by the communities $\mathbf{C}_n^i$, with node (a, i) containing $\deg^i(a)$ unpaired half-edges. Firstly, for each $a \in [n]$, we split the $\deg^i(a)$ half-edges of (a, i) into two distinct groups, which we call *community* half-edges and *background* half-edges. For $a \in \mathbb{Z}$ and $b \in [0, 1]$ define the random variable $[a + b]$ as

$$[a + b] = \begin{cases} a & \text{with probability } 1 - b, \text{ and} \\ a + 1 & \text{with probability } b. \end{cases}$$

(Note that $\mathbb{E}[a + b] = a(1 - b) + (a + 1)b = a + b$.) Now define $Y_a = \lfloor (1 - \xi) \deg^i(a) \rfloor$ and $Z_a = \deg^i(a) - Y_a$ (note that Y_a and Z_a are random variables with $\mathbb{E}[Y_a] = (1 - \xi) \deg^i(a)$ and $\mathbb{E}[Z_a] = \xi \deg^i(a)$ and since we generate each layer separately they are different for each layer) and, for all $a \in [n]$, split the $\deg^i(a)$ half-edges of (a, i) into Y_a community half-edges and Z_a background half-edges. Next, for all $j \in [L^i]$, construct the *community graph* $G_{n,j}^i$ as per the configuration model (see Section 4.1) on node set C_j^i and degree sequence $(Y_a, a \in C_j^i)$. Note that C_0 consists of inactive nodes which, by design, have degree zero. Hence, there is no need to do anything with them. Finally, construct the *background graph* $G_{n,0}^i$ as per the configuration model on node set $[n]$ and degree sequence $(Z_a, a \in [n])$. In the event that the sum of degrees in a community is odd, we pick a maximum degree node (a, i) in said community and replace Y_a with $Y_a + 1$ and Z_a with $Z_a - 1$. Note that $G_{n,j}^i$ is a graph, and C_j^i is the set of nodes in this graph; we refer to C_j^i as a *community* and $G_{n,j}^i$ as a *community graph*. Note also that $G_n^i = \bigcup_{0 \leq j \leq L^i} G_{n,j}^i$.

4.3.5. Phase 5: rewiring self-loops and multi-edges

We continue concentrating on a given layer $i \in [\ell]$. Note that, although we are calling $G_{n,j}^i$ ($j \in \{0\} \cup [L^i]$) *graphs*, they are in fact *multi-graphs* at the end of Phase 4. To ensure that G_n^i is simple, we perform a series of *rewirings* in G_n^i . A rewiring takes two edges as input, splits them into four half-edges, and creates two new edges distinct from the input. We first rewire each community graph $G_{n,j}^i$, $j \in [L^i]$, independently as follows.

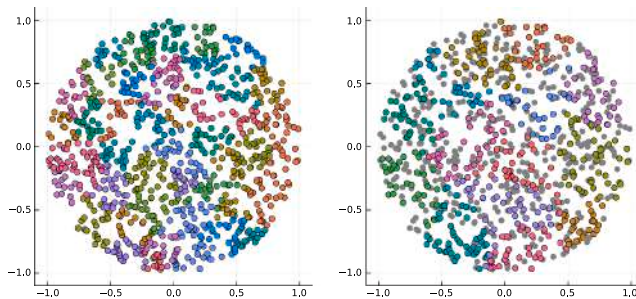


Fig. 4. Two partitions generated based on the same reference layer with $n = 1000$ nodes: (left) $q_1 = 1$ (all nodes active), $S_1 = 32$, $s_1 = 16$, $\beta_1 = 1.5$; (right): $q_2 = 0.5$ (50 % nodes active), $S_2 = 50$, $s_2 = 25$, $\beta_2 = 1.5$ with inactive nodes shown in grey.

1. For each edge $e \in E(G_{n,j}^i)$ that is either a loop or contributes to a multi-edge, we add e to a *recycle* list that is assigned to $G_{n,j}^i$.
2. We shuffle the *recycle* list and, for each edge e in the list, we choose another edge e' uniformly from $E(G_{n,j}^i) \setminus \{e\}$ (not necessarily in the *recycle* list) and attempt to rewire these two edges. We save the result only if the rewiring does not lead to any further self-loops or multi-edges, otherwise we give up. In either case, we then move to the next edge in the *recycle* list.
3. After we attempt to rewire every edge in the *recycle* list, we check to see if the new *recycle* list is smaller. If yes, we repeat step 2 with the new list. If no, we give up and move all of the “bad” edges from the community graph to the background graph.

We then rewire the background graph $G_{n,0}^i$ in the same way as the community graphs, with the slight variation that we also add edge e to *recycle* if e forms a multi-edge with an edge in a community graph or, as mentioned previously, if e was moved to the background graph as a result of giving up during the rewiring phase of its community graph. At the end of Phase 5, we have a simple graph G_n^i representing the i th layer of a multilayer network.

4.3.6. Phase 6: correlations between edges in various layers

During this last phase, we continue performing a series of rewiring (in batches) with the goal of creating a multilayer network with the correlations between edges in various layers (as defined in Section 3.1.3) to be as close to the desired matrix $\mathbf{R}$ (provided as one of the parameters of the model) as possible. It is important to highlight the fact that during this phase, not only do the degrees of the involved nodes not change, but the community degrees stay the same (as well as the background ones). Hence, in particular, the level of noise stays the same.

We run t independent *batches* of operations (by default, $t = 100$). Before every batch, we re-compute the (empirical) correlation matrix $\hat{\mathbf{R}}$ for the current multilayer network ($G_n^i : i \in [\ell]$) and compare it with the desired matrix $\mathbf{R}$. We select an entry ij at random with the probability proportional to the discrepancy between the empirical and the desired values. In other words, we select a pair (i, j) ($1 \leq i < j \leq \ell$) with probability

$$p_{ij} = \frac{|\hat{r}_{ij} - r_{ij}|}{\sum_{1 \leq r < s \leq \ell} |\hat{r}_{rs} - r_{rs}|}.$$

We attempt to rewire $\lceil \epsilon \min\{|E_i^j|, |E_j^i|\} \rceil$ of edges in each batch with the goal to bring $\hat{r}_{ij}$ closer to r_{ij} (by default, $\epsilon = 0.05$). Recall that E_i^j can be viewed as the set of edges in layer i that are between actors that are active in layer j (and, trivially, also active in layer i since inactive actors form isolated nodes), see Eq. (1).

Suppose first that $\hat{r}_{ij} < r_{ij}$, that is, the correlation between layer i and layer j is smaller than what we wished for. Each of the attempts does the following. Randomly select one of the two graphs, G_n^i or G_n^j , and call it *primary*. Then, pick a random edge uv from the primary graph between actors that are active in both layers. Our goal is to try to introduce edge uv in the other graph (call it *secondary*) unless it is already there, in which case we simply finish this attempt prematurely. If u and v belong to one of the communities in the secondary graph (say to the community C), then we take u' to be a random neighbour of u in C (if there are any), take v' to be a random neighbour of v in C (again, if there are any). If u, v, u', v' are four distinct nodes and there is no edge $u'v'$ in the secondary graph, then we remove the two edges uu' and vv' and introduce two new edges uv and $u'v'$. If anything goes wrong, then we simply finish prematurely and move on to another attempt. If u and v are from two different communities in the secondary graph, then the procedure is exactly the same, but this time, our goal is to select four nodes, each from a different community. Specifically, we try to pick a random neighbour u' of u outside of the communities u or v belong to. Then, we try to pick a random neighbour v' of v outside of the communities u, v , or u' belong to. If the four selected nodes are different and there is no edge $u'v'$ in the secondary graph, we do the rewiring.

Suppose now that $\hat{r}_{ij} > r_{ij}$, that is, the correlation between layer i and layer j is larger than what we wished for. As before, during each attempt, we randomly make one of the two graphs, G_n^i, G_n^j , to be *primary* and the second one to be *secondary*. Then, pick a random edge uv from the intersection of the two graphs. Our goal is to try to remove edge uv from the secondary graph. If u and v belong to one of the communities in the secondary graph (say to the community C), then we take a random edge $u'v'$ from C . If u, v, u', v' are four distinct nodes and there are no edges uu' nor vv' in the secondary graph, then we remove the two edges uv and $u'v'$ and introduce two new edges uu' and vv' . As before, if anything goes wrong, then we simply finish prematurely and move on to another attempt. If u and v are from two different communities in the secondary graph, then we pick a random edge $u'v'$ from the secondary graph with the property that all four nodes belong to different communities. We try to rewire the two edges, making sure that no multi-edges get created.

The goal of the sequence of t batches is to bring the (empirical) correlation matrix $\hat{\mathbf{R}}$ closer to the desired matrix $\mathbf{R}$. Unfortunately, fixing one entry of $\mathbf{R}$ may affect the other entries. Hence, it is not guaranteed that the best solution is found after exactly t batches. To take this into account, we track the quality of the multilayer networks ($G_n^i : i \in [\ell]$) at the beginning of each batch (via L_2 norm between $\hat{\mathbf{R}}$ and $\mathbf{R}$) and the final network is the one of the t networks that performed best.

4.3.7. Implementation

The algorithm is implemented in Julia programming language, a high-level, general-purpose dynamic programming language, designed to be fast and effective. Source code and documentation are available on **mABCD** GitHub repository.⁶ The Python ports are available as well.⁷

5. Properties of the mABCD model

In this section, to justify **mABCD** model design and to highlight important features, we provide various experiments to investigate some important properties of the model.

5.1. Correlations between node degrees

Recall that in Phase 2 of the **mABCD** model, independently for all layers, each active node associated with an actor $a \in [n]$ independently generates a normally distributed random variable $X_a = N(a/n, \sigma)$ for some fixed value of the variance σ . In the first experiment, we independently generate the degree distributions for various values of $\sigma \in [0, 20]$. We compute the Kendall rank correlation coefficient between actors' labels a and the corresponding random variables X_a .

To see how quickly the process converges, we compare the results for the number nodes n in $\{10^3, 10^4, 10^5, 10^6\}$ and all actors being active—see Fig. 5 (Left). To see how close the coefficients are to their asymptotic limit (as $n \rightarrow \infty$), we also look at the difference between the values for $n \in \{10^3, 10^4, 10^5\}$ and the ones for $n = 10^6$ —see Fig. 5 (Right). The differences are rather small, even for n as small as 1000. Of course, there are some natural fluctuations, but they are rather insignificant compared to the expectation. It justifies our design of the algorithm. In practice, based on the experiments with $n = 10^6$ (that are saved and accessible by the algorithm), one can select an appropriate value of the parameter σ that is expected to produce the desired Kendall rank correlation coefficients τ . With such a choice of σ , generating 20 independent sequences is enough to make sure that one of them is very close to the desired value of τ .

In the second experiment, we generate the degree sequences for $n = 1000$ actors in 5 layers. The desired correlations between the actors' labels and the order of the corresponding random variables X_a is fixed to be $(\tau_i) = (1.0, 0.5, 0.0, -0.5, -1.0)$ and not all the actors are active in all

⁶ <https://github.com/Krainiski/MLNABCDGraphGenerator.jl>

⁷ <https://github.com/anty-filidor/spreading-vs-mln-structure>

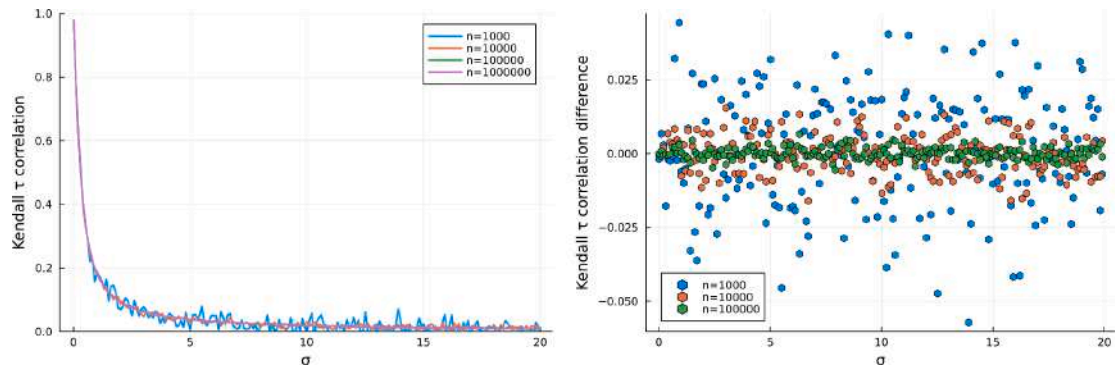


Fig. 5. Left: Kendall rank correlation coefficients between actors' labels and generated random variables $X_a = N(a/n, \sigma)$ for various values of n . Right: Difference between Kendall rank correlation coefficients for small values of n and the ones for $n = 1,000,000$.

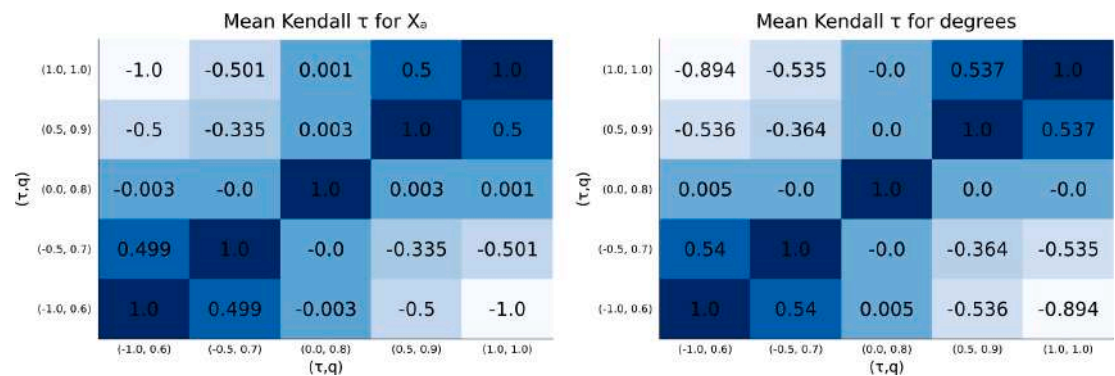


Fig. 6. The average value of Kendall τ rank correlation (over 100 repetitions) between sequences of random variables X_a (Left) and degree sequences (Right) in two layers. Correlations are computed on nodes that are active in both layers.

layers: $(q_i) = (1.0, 0.9, 0.8, 0.7, 0.6)$. The degree sequence for active nodes is independently generated in each layer with the following parameters: $\gamma_i = 2.5$, $\delta_i = 5$, $\Delta_i = 50$.

We repeat the above experiment 100 times. The experimental means and standard deviations of the corresponding Kendall τ correlations between the hidden labels of active actors and generated random variables X_a are $(1.0, 0.5, 0.0, -0.5, -1.0)$ and, respectively, $(0.0, 0.001, 0.002, 0.001, 0.0)$, very close to the desired sequence $(1.0, 0.5, 0.0, -0.5, -1.0)$. We also computed correlations between sequences of X_a as well as degree sequences in all $\binom{5}{2}$ pairs of layers (on nodes that are active in both layers). The experimental means are reported in Fig. 6 for sequences of X_a (Left) and for degree sequences (Right). As expected, both matrices contain similar values, and correlations behave as expected. For example, layers 4 and 5 are negatively correlated with the labels of actors $(-0.5$ and -1.0 , respectively) and so are positively correlated with each other $(0.5$ and 0.537 in the two corresponding correlations).

The third and final experiment in this subsection is merely a sanity check, as this property is enforced in the model by design. We generate degree sequences for three layers with $n = 100,000$ actors in each, all of them being active ($q = 1$). The minimum and the maximum degrees are fixed to be $\delta = 5$, $\Delta = 316 \approx \sqrt{n}$ across the three layers, but the power-law exponents vary: $(\gamma_i) = (2.2, 2.5, 2.8)$. For a given integer k , let $f(k)$ be the experimental cumulative degree distribution, that is, $f(k)$ is the fraction of nodes of degree at least k . For a given set of parameters, the theoretical cumulative degree distribution is equal to

$$\hat{f}(k) = \frac{\sum_{i=k}^{\Delta} i^{-\gamma}}{\sum_{i=\delta}^{\Delta} i^{-\gamma}}.$$

We show that the experimental degree distributions are very close to the desired, theoretical, ones—see Fig. 7 (Left) for the cumulative degree distributions of the three layers.

5.2. Correlations between partitions

Recall that in Phase 4 of the **mABCD** model, active nodes in each layer are independently partitioned into communities of given sizes. The sequences of community sizes can be substantially different across layers, and the process itself is random. However, the underlying geometry of the reference layer (that is used for all of the layers) ensures that the generated partitions are correlated.

In the first experiment, we verify how different values of the parameter β (power-law exponent for the community sizes) affect the **AMI** between two partitions, the first one generated with parameter β_1 and the second one generated with parameter β_2 —see Fig. 8. The difference is very small, with a slightly larger correlation obtained for small values of β . We also check how the choice of the dimension d affects the **AMI**. Not surprisingly, a larger correlation is obtained for dimension $d = 1$, but the 2-dimensional reference layer still produces strong correlations.

We also experiment with the parameter q (fraction of active nodes) and obtain similar conclusions—see Fig. 9. It seems that the **AMI** slightly decreases as $\min\{q_1, q_2\}$ decreases. The **AMI** is more sensitive with respect to parameter S (the upper bound for community sizes)—see Fig. 10. As expected, the correlation decreases in highly imbalanced scenarios; one of the partitions has many small communities, whereas the second one has a few large ones.

In the second experiment, we check how the correlation strength parameter r affects the **AMI** between the two layers—see Fig. 11. It was

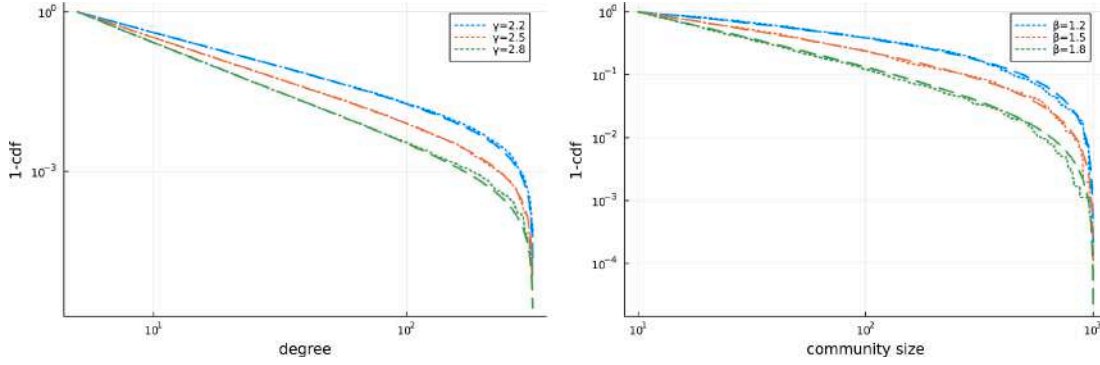


Fig. 7. Left: Empirical (dots) and theoretical (dashes) log-log cumulative degree distributions for three degree sequences: $n = 100,000$, $\delta = 5$, $\Delta = 316$. Right: Empirical (dots) and theoretical (dashes) log-log cumulative community sizes distributions for three sizes sequences: $n = 100,000$, $s = 10$, $S = 1000$.

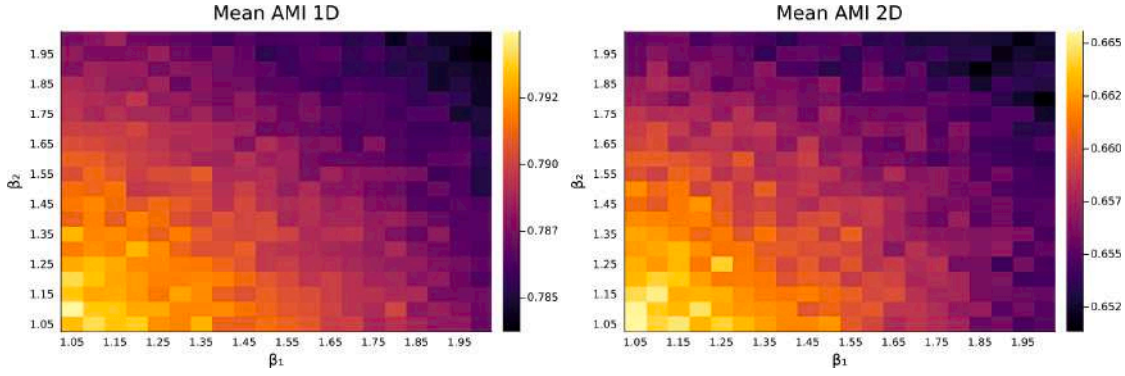


Fig. 8. The average value of AMI (over 500 repetitions) between two partitions generated with β_1 and, respectively, β_2 : $n = 1000$, $s_1 = s_2 = 8$, $S_1 = S_2 = 32$; d -dimensional reference layer was used: (left) $d = 1$, (right) $d = 2$.

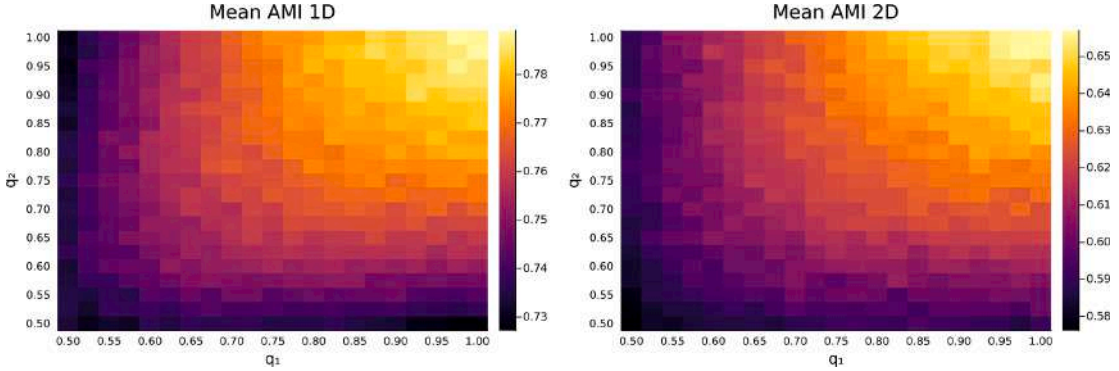


Fig. 9. The average value of AMI (over 100 repetitions) between two partitions generated with q_1 and, respectively, q_2 : $n = 1000$, $s_1 = s_2 = 8$, $S_1 = S_2 = 32$, $\beta_1 = \beta_2 = 1.5$; d -dimensional reference layer was used: (left) $d = 1$, (right) $d = 2$. Only common active nodes are kept in both layers.

introduced to provide a smooth transition between the maximum possible correlation (with $r = 1$) and completely random partitions (with $r = 0$). Our experiments confirm this desired property and show that the AMI is quite stable even for small networks on $n = 1000$ nodes.

In the third experiment, we generate partitions of $n = 1000$ actors in 5 layers. As before, not all the actors are active in all layers: $(q_i) = (0.6, 0.7, 0.8, 0.9, 1.0)$. We test different values for the distribution of the community sizes: $(\beta_i) = (1.1, 1.3, 1.5, 1.7, 1.9)$, $(s_i) = (8, 24, 40, 56, 72)$, and $(S_i) = (32, 48, 64, 80, 96)$. More importantly, we test different values of the correlation strength between communities and the reference layer:

$(r_i) = (0.0, 0.25, 0.5, 0.75, 1.0)$. The experiment is repeated 100 times, and the experimental means of the AMI between $\binom{5}{2}$ pairs of partitions are reported in Fig. 12. As expected, the partition of active actors in the first layer (with $r_1 = 0.0$) is not correlated with any other partition. Moreover, the values of AMI increase together with the corresponding correlation strengths.

In the fourth and final experiment in this subsection, we generate sequences of community sizes with different distributions. There are $n = 100,000$ actors in each of the three layers, all of them being active ($q = 1$). The minimum and the maximum community size is fixed to be

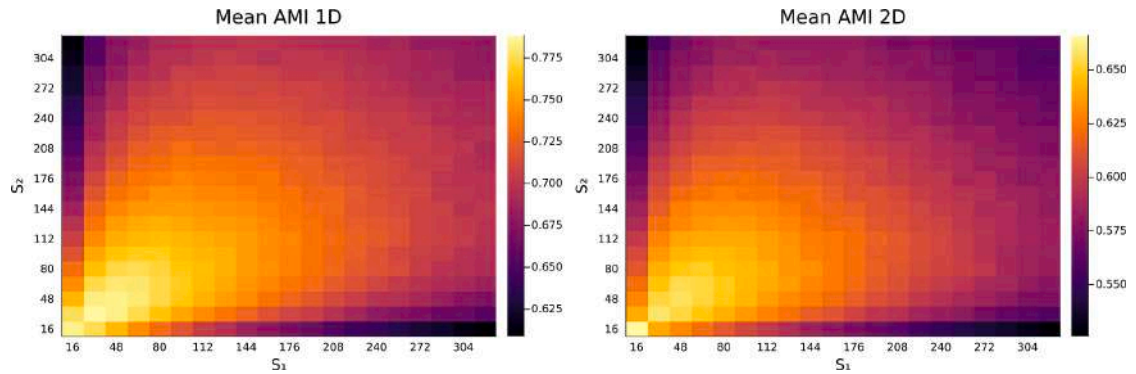


Fig. 10. The average value of AMI (over 500 repetitions) between two partitions generated with S_1 and, respectively, S_2 : $n = 1000$, $s_1 = s_2 = 8$, $\beta_1 = \beta_2 = 1.5$; d -dimensional reference layer was used: (left) $d = 1$, (right) $d = 2$.

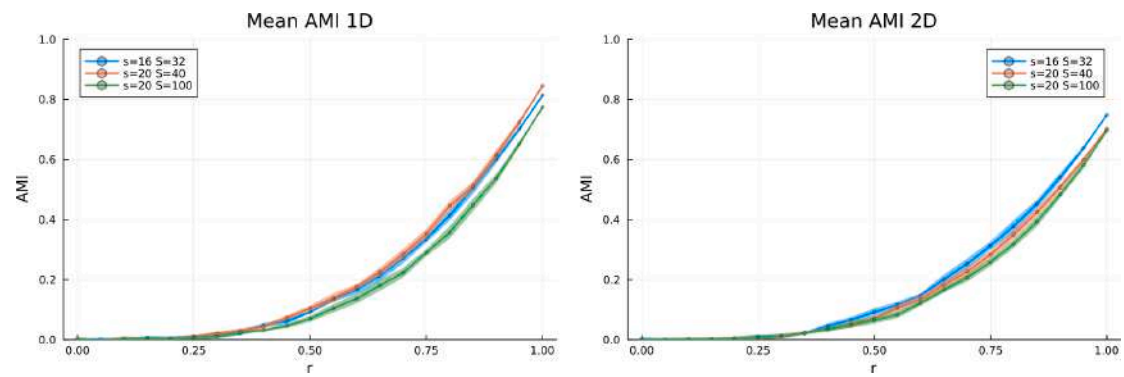


Fig. 11. The average value of AMI ± 1 standard deviation (over 10 repetitions) between two partitions generated with different values of r (x-axis): $n = 1000$, $\beta_1 = \beta_2 = 1.5$; d -dimensional reference layer was used: (left) $d = 1$, (right) $d = 2$.

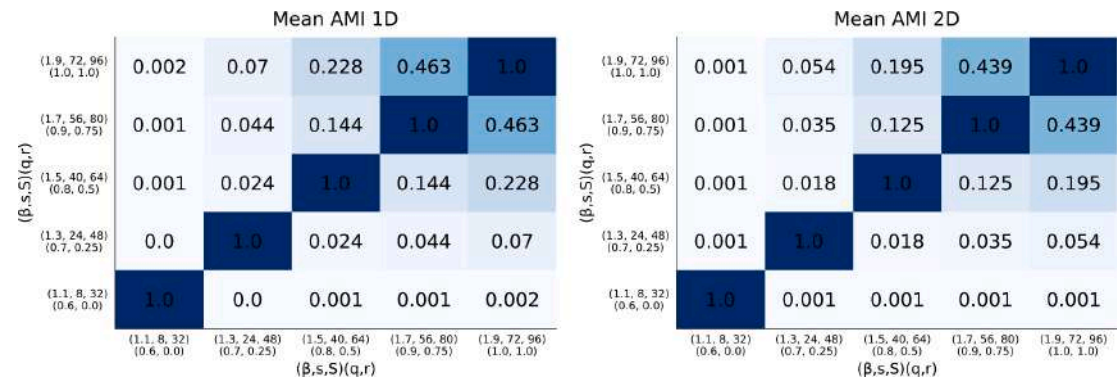


Fig. 12. The average AMI value between pairs of partitions and in generated **mABCD** models as a function of β , s , S , q , r (over 100 repetitions); d -dimensional reference layer was used: (left) $d = 1$, (right) $d = 2$. Only common active nodes are kept in compared layers.

$s = 10$ and $S = 1000$ across the three layers, but the power-law exponents vary: $(\beta_i) = (1.2, 1.5, 1.8)$. We show that the experimental community sizes distributions are very close to the desired ones—see Fig. 7 (Right) for the cumulative community sizes distributions of the three layers.

5.3. Correlations between edges in various layers

In this subsection, we investigate the performance of Phase 6 of the process generating **mABCD**. Recall that in this phase, in each of the t batches, we carefully rewire a random fraction of edges in one of the

layers with the goal to bring the empirical correlation matrix $\hat{\mathbf{R}}$ closer to the desired matrix $\mathbf{R}$. However, because of the very rich dependence structure between layers (associated with various objects: degree sequences, partitions into communities) and the fact that some nodes are inactive, it is not clear that the desired correlations can be achieved.

We selected 4 real-world networks (ckmp, l2-course, lazega, and timik), extracted their properties (such as degree sequences, community sizes sequences, noise ratio, correlations; see Section 6.2 for more details on how it can be done), and created corresponding **mABCD** networks mimicking these real-world networks. To provide a good fit to real-world networks we injected exact degree sequences (and therefore

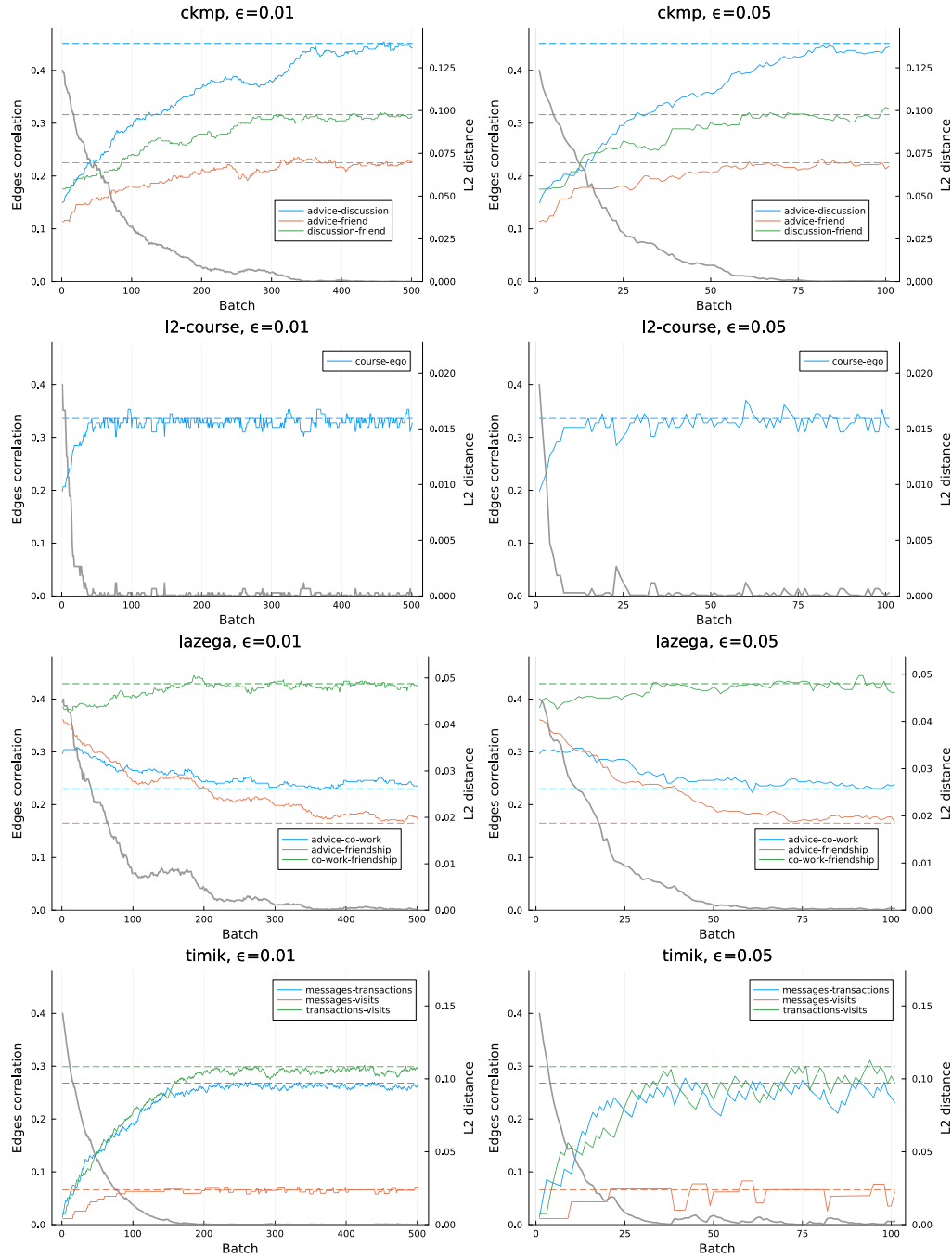


Fig. 13. Changes in edge correlation between layers (solid) compared to the desired correlation (dashed). Grey thick line represents the L_2 distance between the desired and the empirical correlation matrices ($\|\mathbf{R} - \hat{\mathbf{R}}\|_2$).

active nodes) into the algorithm. As the ground-truth community structure is not known, we used Louvain algorithm to generate community sizes sequence, estimated power law exponent for each layer, and used extrema of community sizes as input to synthetic graph. The convergence to the desired correlation matrix $\mathbf{R}$ between edges in various layers is presented in Fig. 13. The default parameters for the number of batches ($t = 100$) and the fraction of edges rewired ($\epsilon = 0.05$) were

tested (right column) but we also checked other set of parameters with the same total number of rewires, namely, $t = 500$ and $\epsilon = 0.01$.

There is no visible difference between the two scenarios which justifies the default parametrization: the smaller number of batches yields a faster algorithm as the algorithm needs to recompute the empirical correlation matrix $\hat{\mathbf{R}}$ at the beginning of each batch. In general, the process converges quite quickly: the L_2 norm between $\hat{\mathbf{R}}$ and $\mathbf{R}$ decreases

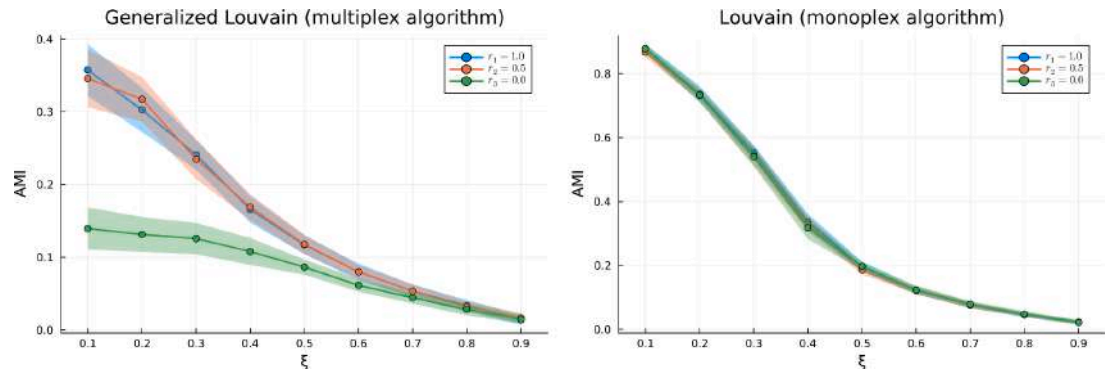


Fig. 14. The average (± 1 std) Adjusted Mutual Information between the ground truth and a) Generalized Louvain's communities (left), b) Louvain's communities (right); 50 repetitions for each ξ . (Note that for Louvain's communities all three lines overlap; for Generalized Louvain lines for $r_1 = 1.0$ and $r_2 = 0.5$ are almost overlapped).

rapidly. However, there are some fluctuations and the final network is not necessarily the best one. This, again, justifies the design of the algorithm that keeps the best network generated during the process.

5.4. Community detection in **mABCD** networks

As noted in Section 1, the **mABCD** model supports the validation and assessment of community detection algorithms. Its generation of ground-truth communities enables direct comparison with partitions produced by tested methods. As an extension of **ABCD**, **mABCD** incorporates findings from prior analyzes of monoplex community detection algorithms (Kamiński et al., 2022, 2021a). An open question remains regarding its utility for validating multiplex partitioning algorithms (Magnani et al., 2021a). Although relevant, this issue lies beyond the scope of this study. We therefore conduct a simple validation experiment using one multiplex partitioning algorithm and the monoplex counterpart, leaving broader analysis for future work.

We generated a network with 1000 actors and three layers. Configuration of layers was identical ($q = 1, \tau = 0.0, \gamma = 2.5, \delta = 2, \Delta = 25, \beta = 1.5, s = 16, S = 64$) except for the community correlation coefficient r ($r_1 = 1.0, r_2 = 0.5, r_3 = 0.0$). The edges correlation matrix $\mathbf{R}$ was not defined, hence no edges rewiring occurred. The noise ratio ξ varied from 0.1 to 0.9 to simulate increasing inter-community connectivity. Due to the stochastic nature of **mABCD**, ten graphs were produced for each ξ . Community detection was performed using the Generalized Louvain algorithm (Mucha et al., 2010) on the whole multiplex network and the base Louvain algorithm (Blondel et al., 2008) on each layer independently. The resulting partitions were compared with the ground truth from **mABCD** via the AMI metric. The experiment aimed to examine how the quality of group detection depends on two parameters: the fraction of inter-community edges within layers (ξ) and the community alignment across layers (r). The Generalized Louvain algorithm is capable of detecting *pillar communities* (Magnani et al., 2021a). One can think about pillar communities as groups comprising the same nodes that consistently emerge across the network layers. Therefore, **mABCD** communities in layers with high r (correlated with latent layer, which implies that also between themselves) should create higher-level structures possible to detect with multiplex algorithm. Generalized Louvain highlights pillar communities by producing consistent clusters' labels for nodes in all layers. For reporting purposes, we compare the clustering for each layer separately (Fig. 14).

Aligned with expectations, the results show a decline in clustering quality with increasing noise. Clustering within each layer converges to random partitioning for high ξ values. Notably, Generalized Louvain exhibits curious tendency related to the r parameter. The method performed equally well for perfectly correlated ($r_1 = 1.0$) and moderately

correlated ($r_2 = 0.5$) layers (the reason is that the algorithm does not see the latent layer, so both layers are indistinguishable from the perspective of community detection, since the third layer is uncorrelated). Performance for the uncorrelated layer is significantly lower, which is expected as the assignment of actors to communities is randomly shuffled, so the signal coming from it to the Generalized Louvain algorithm is weaker. Captured similarities may indicate the ability of Generalized Louvain to detect intra-layer patterns that affect the structure of communities. In contrast, the performance of the Louvain algorithm is much better than the multiplex extension. The monoplex variant is insensitive to the value of r , which is aligned with **mABCD** design. The r parameter controls correlation between layers, but does not affect strength of communities within layers—this property is controlled by the ξ parameter.

5.5. Computational efficiency of the model

One of the reasons why the original **ABCD** model was developed was the fact (pointed out by the industry partners we worked with back then) that other synthetic random graph models with community structure do not perform well in terms of computational efficiency. Keeping this concern in mind, the design of the **ABCD** model aims to generate networks fast. Indeed, after careful preprocessing (that is very fast), **ABCD** is essentially a union of independent copies of the configuration model. In two minutes, on a typical laptop, one can generate networks on $n = 1.5 \times 2^{21}$ nodes. The **mABCD** model is clearly much more complicated than the original **ABCD**, but it builds on fast ingredients of **ABCD**. Therefore, in order to validate whether it preserves computational advantages inherited from its monoplex counterpart, we performed experiments aimed to address this problem. The data reported below was obtained during simulations executed on Macbook Air M1 (2020) with macOS v15.5.

Not surprisingly, Phase 6 (switching edges to get the desired correlations between edges in various layers) is the most time-consuming phase. Moreover, it is clear that the more layers, the longer the process is. In two minutes, one can generate **mABCD** networks on $n = 2^{16} = 65,536$ nodes. However, without Phase 6, in the same time one can produce much larger networks on $n = 2^{20} = 1,048,576$ nodes—see Fig. 15. The percentage breakdown is presented in Fig. 16.

Finally, we have compared the speed of **mABCD** model to the **multilayerGM** framework proposed by Bazzi et al. (2020) (see Section 2.1 for more details). The results are presented in Fig. 17 and Table 4, and indicate that for the same setup, **mABCD** is roughly 10 times faster for small networks (1,024 nodes), and the difference in speed quickly increases with the network size (for a two-layer network with 32,768 nodes **mABCD** is 310 times faster). Since generating one five-layer net-

Table 4

The average execution time (in seconds) of **multilayerGM** (first) and **mABCD** (second) algorithms for varying number of nodes n and number of layers ℓ ; ratio of execution times in brackets.

n	$\ell = 2$	$\ell = 3$	$\ell = 4$	$\ell = 5$
1,024	0.65/0.03 (22)	0.98/0.06 (16)	1.31/0.11 (12)	1.67/0.17 (10)
2,048	2.44/0.07 (33)	3.71/0.16 (24)	4.84/0.27 (18)	6.00/0.43 (14)
4,096	11.01/0.22 (50)	16.30/0.46 (36)	21.83/0.80 (27)	27.27/1.26 (22)
8,192	55.46/0.62 (90)	82.99/1.29 (64)	110.85/2.33 (48)	137.93/3.72 (37)
16,384	274.03/1.73 (158)	410.23/3.69 (111)	545.92/6.70 (82)	681.25/10.59 (64)
32,768	1,533.2/4.9 (310)	2,290.1/10.5 (219)	3,051.3/18.6 (164)	3,817.6/29.2 (131)

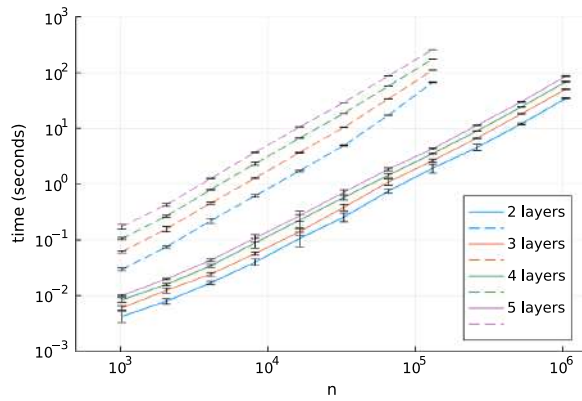


Fig. 15. The average (± 1 std) execution time of **mABCD** (in log-log scale) with Phase 6 (dashes) and without it (solid) for $n = 2^k$ nodes, $k \in \{10, 11, \dots, 20\}$, and $\ell \in \{2, 3, 4, 5\}$ layers; 10 repetitions for each combination.

work with 32,768 nodes took more than one hour for **multilayerGM**, and we repeated experiments ten times for each combination of node and layer, we stopped our comparison there.

6. Using mABCD for spreading phenomena analysis

To demonstrate the applicability of the **mABCD** synthetic network in relevant fields, we present three use cases from the area of spreading phenomena analysis. These experiments aim to illustrate the usefulness of the introduced framework and to provide guidance for conducting similar experiments. In general, the problem which we aim to tackle is how the topological features of the network affect the spread of influence in multilayer networks. To do so, we are using **mABCD** to generate various networks where one feature changes while the remaining ones remain unchanged. This allows us to assess how a single feature affects the spreading process. It is worth noting that the problems depicted below can also be regarded as standalone research outlines within a broader context beyond the examples presented here. Before describing the experiments, we first discuss the employed methodology, as they are conducted in a similar manner.

6.1. Scope and methodology

The experimental part consists of several steps. First, we select a real-world network to serve as a baseline; specifically, we decided to use the *timik* network (see Table 1), due to its large size which helps minimize sampling errors (typical to any graph generator). Then, a set of configuration parameters for this network are estimated. Having done so, we are able to manipulate specific parameters and generate synthetic networks that can be considered derivatives of the original network. It is worth noting that the idea of utilizing a real network stems from the need to bring the experiments closer to real-world conditions. In the context of

spreading phenomena, one can imagine that, as the baseline network represents a real-world system, its artificially generated variants using the **mABCD** model can correspond to possible configurations of such a system, resembling the impact of certain policies imposed on it. For instance, since the employed network (*timik*) represents interactions on an online social platform, its artificial counterpart generated with an adjusted Δ_i parameter may reflect the effect of a policy that regulates the maximal number of friends allowed. Thus, by comparing spreading on the original network and its derivative, we assess the consequences of counterfactual scenarios that affect the construction of the evaluated system.

To ensure a fair comparison of spreading effectiveness, only graphs generated by **mABCD** are evaluated. Consequently, instead of simulating spread on the original real-world network, its “digital twin” is employed. Based on the estimated configuration parameters (adjusted for minimum degree and minimum community size to remain within the capabilities of the framework), 20 instances of the network are generated using **mABCD**. Spread is then simulated on generated networks under a selected range of spreading regimes that produce non-trivial dynamics; that is, the network neither saturates too rapidly nor inhibits the initiation of spreading.

Each experiment involves manipulating a single parameter of the **mABCD** model, with the aim of evaluating networks that differ from the baseline by only that feature. Similarly to the baseline graph, we generated 20 multilayer networks for each set of configuration parameters to mitigate the randomness inherent in **mABCD**. Diffusion is then simulated on these networks using the same spreading conditions as for the “digital twin” of the real-world network. Since the employed spreading model is also nondeterministic, each simulation was repeated 30 times to reduce the influence of random effects. Finally, the diffusion effectiveness observed in the modified and original networks is compared to assess the influence of the parameter under investigation on the spreading process.

As the influence spreading mechanism, we utilize the Independent Cascade Model (Goldenberg et al., 2001) (ICM), extended to the multilayer scenario (MICM). This adaptation draws upon a similar rationale to that used in extending the related Linear Threshold Model (LTM), as described by Zhong et al. (2022). Consistent with their approach, we regard actors as the principal entities in the diffusion process, while their corresponding nodes across various layers serve a supplementary function. In the original ICM, the probability that an active agent will activate its inactive neighbour is controlled by a single parameter, π , with each activation attempt occurring only once. Our modification introduces an additional parameter, δ , termed the *protocol function*, which dictates how activation signals from different layers are combined to establish the actor’s overall state. Activation of the actor (and all its layer-specific nodes) occurs only if the protocol condition is fulfilled; if not, the actor remains inactive even if some layer-specific nodes receive positive activation signals. We examine two extreme cases for δ : the *AND* protocol, where activation requires positive signals from *all* layers representing the actor; and the *OR* protocol, where activation in *any* single layer suffices.

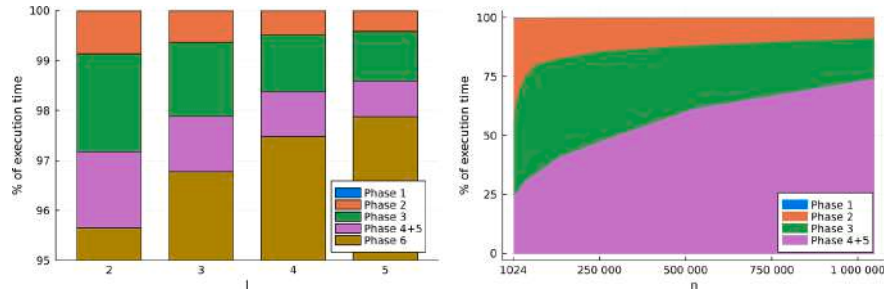


Fig. 16. Percentage breakdown of the execution time between phases of the algorithm: with Phase 6 included $n = 2^{16}$ (Left) and with Phase 6 excluded, $\ell = 5$ (Right).

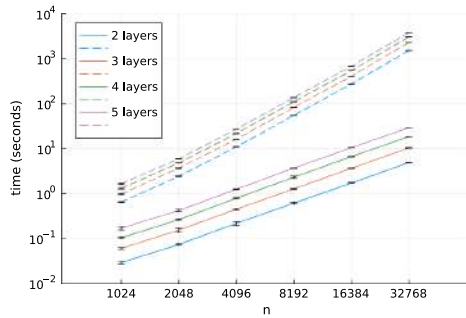


Fig. 17. The average (± 1 std) execution time (in log-log scale) for **multilayerGM** framework (dashes) and **mABCD** model (solid) for $n = 2^k$ nodes, $k \in \{10, 11, \dots, 15\}$, and $\ell \in \{2, 3, 4, 5\}$ layers; 10 repetitions for each combination.

At this point it is worth noting that, in contrast to the ICM, employing its well-known counterpart in the experimental part, the LTM (Granovetter, 1978), can lead to unstable results. This is due to the way **mABCD** generates graphs, namely, it aims to capture community structure (i.e., global properties) rather than local ones. As a result, graphs generated with **mABCD** contain groups of nodes that induce denser subgraphs compared to the global density, but locally neighbourhoods still resemble tree structure, as there are very few short cycles. By contrast, real networks do not exhibit this local behaviour; they contain many short cycles, even triangles. Consequently, diffusion in **mABCD**-generated networks under the LTM, which relies on a threshold mechanism requiring a certain number of active neighbours to activate a given node, can either quickly die out or abruptly saturate the network, with no parameter range that produces gradual, stable spreading. On the other hand, the mechanism inherent to the ICM is “immune” to this tree-like local structure, as it focuses on individual edges rather than the entire neighbourhood of a given node.

Finally, the metric used to compare the effectiveness of spreading was *gain* (Czuba & Bródka, 2025). It quantifies the total network coverage achieved by initiating the diffusion from a seed set S_0 . Specifically, it is computed as the proportion of actors activated during the diffusion process, excluding the initial seeds, relative to the total number of activatable actors (i.e., those not in S_0):

$$\Gamma = \frac{|S_\infty - S_0|}{|A - S_0|}, \quad \Gamma \in [0, 1].$$

In summary, the experimental methodology adopted in this study comprises the following steps: (1) selecting a real-world network; (2) obtaining a set of configuration parameters of the model for the network; (3) generating 20 network instances serving as “digital twins” of the real-world network; (4) simulating diffusion under a predefined regime; (5) manipulating a specific parameter of the configuration model; (6) generating 20 network instances based on the modified configuration;

(7) simulating diffusion under the same predefined regime; (8) comparing the diffusion dynamics to assess how the evaluated parameter influences spreading efficiency.

The simulations described in this chapter were conducted on a workstation running Ubuntu 20.04.4 LTS with kernel version 6.5.3-arch1-1, equipped with 376 GB of RAM and an Intel(R) Xeon(R) Gold 6238 CPU @ 2.10 GHz (x86_64 architecture). The source code was written in Python 3.12 utilizing **mABCD** v1.0, **networkx** v3.3 (Hagberg et al., 2008), and **network-diffusion** v0.18 (Czuba et al., 2024), with particular attention to reproducibility. All scripts, data, and results are publicly available at Github repository.⁸

6.2. Extracting configuration parameters from real networks

To describe how the configuration parameters are retrieved from real-world networks, we first refer the reader to Section 4.2, as the notation used below follows that section closely. Furthermore, we do not elaborate here on how the values of n , ℓ , q , δ , and Δ are obtained, as they are straightforward and can be directly extracted from the processed network.

As for the more complicated parts of the **mABCD** framework, we begin with **R**. This parameter is denoted in a form of matrix of pairwise edge correlations. We estimate it using directly an approach described in Section 3.1.3, especially with Eqs. (1) and (2).

The **mABCD** model assumes that the labels of actors and nodes representing them follow the natural numbers. However, in real networks, labels may take various other forms. This aspect affects the estimation of τ , which relies on the natural ordering of actor labels. To address this in retrieving τ , we first convert the node labels in the first layer (layers are sorted alphabetically) so as to maximize the correlation between node degrees and their assigned labels; that is, the node with the highest degree is assigned the highest identifier. These re-organized labels are then used to compute the correlations in the remaining layers. Moreover, only nodes with a positive degree are taken into account to obtain the correlation value in order to reduce the noise.

The parameter r is the most challenging to retrieve, as we do not have access to the latent biscuit-like reference layer used during the network generation process. To address this, we apply a coarse estimation of this parameter. Specifically, as it was done before, we select the first layer in alphabetical order and apply the Louvain algorithm to identify its community structure, which is then treated as the reference partition. Subsequently, we detect the community structure in each layer of the network and compute the layer-specific value r_i as the AMI between the layer's partition and the reference partition. Since the alphabetically first layer is always used as the reference, its r value will be large. Due to the non-deterministic nature of the Louvain algorithm (Blondel et al., 2008), this value will not be exactly 1, but it should be close to it.

⁸ <https://github.com/anty-filidor/spreading-vs-mln-structure>

Among the degree-oriented properties, only the retrieval of γ is non-trivial, as it involves the general challenge of fitting an observed distribution to a specific model. To estimate this parameter, we employ the `powerlaw` library (Alstott et al., 2014), implicitly assuming that the degree distribution follows a power-law and is not better described by any other distribution type.

The next group of parameters (β , s , and S) concerns layer-specific community structural properties. At this stage, we again employ the Louvain algorithm (Blondel et al., 2008) to obtain a partition of the nodes into communities on each layer of the network. Then, a value of β is estimated by applying the `powerlaw` library (Alstott et al., 2014) to the sequence of community sizes. The values of s and S are likewise derived from this sequence.

Finally, to estimate the layer-wise noise level between communities, we again analyze the community structure on each network layer. For each ξ_i , we compute the fraction of inter-community edges relative to the total number of edges in the given layer.

The implementation of the **mABCD** framework also includes parameters that do not directly reflect properties of the generated network, but instead serve as variables controlling the generation process. In the subsequent experiments, we use a fixed set of such parameters. Specifically, the dimensionality of the latent reference layer is set to $d = 2$. The maximum number of iterations for sampling both degrees and community sizes is set to 1000. For the rewiring step, we use 0.05 as the proportion of edges to be rewired in each batch, and 100 as the number of rewiring batches.

It is worth noting that, like any artificial model, **mABCD** accepts only a feasible range of configuration values. Therefore, after retrieving the parameters for *timik*, we tailored them to fit within the constraints of the framework according to Section 4.2. The code to extract all parameters mentioned above from any network is also present in the previously mentioned GitHub repository, so even inexperienced user can generate a “digital twin” of any real network.

6.3. Experiments

Having introduced the outline of the experimental methodology, we now turn to the description of the experiments. To demonstrate the capabilities of **mABCD**, we selected three use cases, each designed to shed light on a different aspect of the relationship between network geometry and spreading phenomena. Specifically, we evaluate the impact of the noise level between communities, network compression, and inter-layer community correlation.

6.3.1. Effect of noise level between communities

The first experiment conducted aimed to answer the following question: how does the inter-community noise level (ξ) impact information spreading? Intuitively, we expected higher ξ values to facilitate the information dissemination and lower ones to suppress it. Indeed, the lower values of ξ could create “echo chambers”, closed environments where actors are primarily exposed to information and opinions that reinforce their existing beliefs, potentially leading to increased fragmentation. On the other hand, dissemination might still occur through other layers. Hence, the situation for multilayered networks is more complex and interesting compared to single-layer ones.

Following this hypothesis, we generated four types of networks with modified values of ξ as follows: *series 2* with $\xi_i = 1.00$, *series 3* with $\xi_i = 2\xi_{timik}$, *series 4* with $\xi_i = 0.5\xi_{timik}$, and *series 5* with $\xi_i = 0.01$. In other words, two types of networks exhibit extreme inter-community noise levels across all layers, while the remaining two have intermediate values, determined proportionally to those of the baseline. We then selected a spreading regime for the baseline networks (i.e., “digital twins” of *timik*, denoted as *series 1*). Specifically, we used the Neighbourhood Size Discount (Czuba & Bródka, 2025) to select seeds, and considered two sets of parameters for each protocol function: $(\delta, \pi, s) \in \{AND\} \times \{0.15, 0.2, 0.25, 0.3\} \times \{1, 3, 5, 7\}$ and $(\delta, \pi, s) \in$

$\{OR\} \times \{0.03, 0.05, 0.07, 0.09\} \times \{1, 5, 10, 15\}$, where δ and π denote the parameters of the MICM (see Section 6.1), while s refers to the seed set budget, expressed as a percentage of the number of actors constituting a given network.

The results indicate that the impact of the noise level between communities on diffusion effectiveness depends on the spreading parameters. Specifically, all three variables— δ , π , and s —contribute in distinct ways. Therefore, we first briefly discuss the influence of each parameter and subsequently present detailed results for the subset of configurations in which manipulating ξ yields the most pronounced effect. The complete set of results is available in the code repository.

The impact of the protocol function is significant: for the less restrictive one (*OR*), the noise level barely affects the diffusion characteristics (up to 0.0052 of Γ obtained for baseline). The trend is contrary to the initial assumptions—namely, reducing ξ improves spreading. One possible explanation could be that low values of ξ help with spreading information within one community in one layer and multilayer structure is responsible for spreading to other communities. By contrast, the differences observed for $\delta = AND$ were more pronounced: up to an order of magnitude larger than for *OR* (particularly in networks with decreased inter-community noise levels). Nonetheless, at this level of analysis, there was no clear trend regarding the impact of ξ on spreading effectiveness for this protocol function. Results obtained for both protocol functions indicate that higher activation probability (π) corresponds to a reduced impact of ξ , i.e. the results converge towards those of the baseline. Regarding the seed set budget size, for $\delta = AND$, lower budgets lead to larger differences with the baseline results, and the results begin to align with the noise level: higher noise yields greater gain. For *OR*, however, different trends were observed. In this case, for the diffusion triggered with higher budgets, the impact of ξ was more visible, although the relative differences with *series 1* remained still small (up to 0.0062 of baseline *Gain*).

Finally, we turn to the set of parameters for which manipulating ξ produces the most pronounced effects. To this end, we selected following spreading conditions: the lowest evaluated budget for the *AND* protocol and the highest for the *OR* protocol. These configurations are listed in Table 5. As can be observed, the impact of ξ on spreading effectiveness differs significantly between the two diffusion regimes. When the protocol function is more restrictive, increasing the inter-community noise enhances information spread. The trend is even more pronounced when noise is suppressed: in such cases, the diffusion coverage may decrease by up to 25 %. This outcome aligns with our initial intuition. Since the *AND* protocol imposes stricter conditions, it becomes easier to activate a given actor when it is more strongly connected at the global level. Consequently, the process can “reach” more distant nodes and has a greater chance of persisting. In contrast, when the *OR* protocol is applied, the effect of modifying ξ is reversed, albeit considerably weaker. A possible explanation is that, when communities are more clearly separated, the diffusion tends to remain within them initially and only later extends outward. Furthermore, due to the lenient nature of the *OR* protocol, the process is less likely to die out prematurely, which may account for this behaviour.

6.3.2. Effect of changes in network size

With the second experiment, we aimed to address one of the fundamental questions in the domain of social network analysis: is it possible to compress a network in such a way that it preserves its topology while being smaller? Our goal was to examine this issue in the context of spreading phenomena. Specifically, we were interested in whether manipulating the number of actors, while preserving all other structural properties of the network (as dictated by the configuration model underlying **mABCD**), would result in different diffusion characteristics. It is worth noting that the approach adopted in this study can be considered novel, as most existing work reduces graphs using dedicated compression techniques rather than by explicitly employing standalone graph generators (for a comprehensive review, see Sun et al. (2024)).

Table 5

Change in Γ achieved through MICM-driven spreading in networks generated with **mABCD** with varying inter-community noise levels, relative to the results obtained on the baseline network (resembling *timik*).

δ	π	ξ_i				
		0.01	$0.50\xi_{timik}$	ξ_{timik}	$2.00\xi_{timik}$	1.00
AND	0.15	-0.1668	-0.0392	0.0000	0.0352	0.0397
	0.20	-0.2440	-0.0450	0.0000	0.0254	0.0290
	0.25	-0.2339	-0.0183	0.0000	0.0049	0.0041
	0.30	-0.1706	-0.0066	0.0000	0.0017	0.0018
OR	0.03	0.0104	0.0088	0.0000	-0.0120	-0.0154
	0.05	0.0046	0.0033	0.0000	-0.0049	-0.0059
	0.07	0.0020	0.0013	0.0000	-0.0022	-0.0026
	0.09	0.0009	0.0008	0.0000	-0.0009	-0.0010

Table 6

Change in Γ achieved through MICM-driven spreading in networks generated with **mABCD** with varying number of actors, relative to the results obtained on the baseline network (resembling *timik*).

δ	π	n				
		$0.50n_{timik}$	$0.75n_{timik}$	n_{timik}	$1.25n_{timik}$	$1.50n_{timik}$
AND	0.15	-0.0011	0.0016	0.0000	0.0911	0.0949
	0.20	-0.0003	0.0005	0.0000	0.0793	0.0808
	0.25	-0.0001	-0.0000	0.0000	0.0428	0.0432
	0.30	0.0002	0.0001	0.0000	0.0202	0.0202
OR	0.03	-0.0006	-0.0011	0.0000	0.0522	0.0542
	0.05	-0.0002	-0.0003	0.0000	0.0341	0.0353
	0.07	-0.0001	-0.0001	0.0000	0.0231	0.0237
	0.09	0.0001	0.0000	0.0000	0.0163	0.0168

In order to address this question, we generated four series of networks, each differing only in the number of actors, while preserving the remaining configuration model parameters from the baseline. The series were defined as follows: $n = 1.5n_{timik}$ (series 6), $n = 1.25n_{timik}$ (series 7), $n = 0.75n_{timik}$ (series 8), and $n = 0.50n_{timik}$ (series 9). Hence, two of the network types can be considered as upsampled, while the other two represent downsampled versions of the baseline. Regarding the spreading model, we applied the same parameter settings as in the first experiment (see Section 6.3.1).

In contrast to the first experiment, the trends observed in the results are consistent and clear when analyzed with respect to the protocol function, activation probability, and seeding budget. Therefore we move directly to Table 6, which presents these results aggregated over the MICM parameters; nevertheless, grouping by other variables produces similar patterns. The results are shown in the same format as in Section 6.3.1, i.e., as the average relative difference in Γ obtained by each network series with respect to the baseline. It is noteworthy that, since Γ accounts for network size, it is reasonable to compare this metric across diffusion processes conducted on networks with different numbers of actors.

As one can note, for both protocol functions and across all evaluated activation probabilities, downsampling the network does not disturb spreading effectiveness (the differences are minimal). On the other hand, when the network gets upsampled, we observe slight, but non-negligible changes, in the diffusion—the more upsampled the network is, the higher Γ change we observe. However, the disparities are not as big as in the results from Section 6.3.1. At this point, one might wonder why network downsampling does not alter the spreading trend, while upsampling does. A possible explanation lies in the average degree of the evaluated networks, as this parameter has a substantial impact on the effectiveness of diffusion under the MICM. We observe that downsampling has little effect on the average degree, whereas upsampling increases it by approximately 0.9 (from around 20.2 to 21.1). Consequently, in the

upsampled graphs, the diffusion process may be more effective due to this slight increase in connectivity, especially when the seed set is selected using a heuristic that prioritizes central actors rather than selects them at random.

6.3.3. Effect of inter-layer community correlation

The final experiment aimed to evaluate whether spreading is more effective in networks where the communities are strongly correlated. The **mABCD** framework allows control over this property, albeit indirectly, via the r_i parameter. (Recall that this parameter governs the extent to which the communities in layer i are correlated with the latent reference layer, where spatial proximity influences community assignment, i.e., actors located closer together are more likely to be placed in the same group; see Section 4.3.3 for more details). To this end, we intended to simulate a case in which information begins to diffuse from actors who are central in only one of the several types of relationships represented in the network, while manipulating community correlations on the remaining layers. For instance, consider a situation in which a group of department heads returns from an overseas business trip, having contracted a disease. As leaders, they occupy central positions within the company network and communities formed by their teams. Subsequently, the disease begins to spread not only through work-related contacts but also through encounters in other spheres of life. By manipulating the r parameter, we wanted to reflect the extent to which these relationships overlap with the professional one, and to assess whether it influences the effectiveness of spreading.

As in the previous experiments, the networks were also generated according to the configuration derived from *timik*. All of them had the r value on the first layer set to 1.000 in order to comply with the assumed seed selection scenario. However, they differed in the r values on the remaining layers, as follows. Two series were generated to represent stronger community correlations: series 10 with $r_2 = r_3 = 1.000$, and series 11 with $r_2 = r_3 = 0.667$. Additionally, networks with weaker correlations were generated: series 12 with $r_2 = r_3 = 0.333$, and series 13 with $r_2 = r_3 = 0.001$. The spreading conditions applied were the same as in the previous experiments, with one difference in the seed selection method. Specifically, instead of employing a heuristic that considers actors' representations across all network layers, we modified it to consider only the first layer (which is maximally correlated with the latent layer used in network generation). Consequently, the Neighbour Size Discount heuristic was reduced to its baseline version designed for single-layer networks: Degree Centrality Discount (Chen et al., 2009) and with that method a seed set was selected.

After executing the simulations, we obtained the results. As in the previous experiments, we initially analyzed them separately with respect to δ , π , and s . Across all of these parameters, no significant differences in spreading were observed, even though the propagation process could be initiated and sustained for several simulation steps. To maintain clarity, we present the results aggregated over s in Table 7. As can be seen, the relative difference in Γ with respect to the *timik*'s "digital twin" (series 1) remains consistently very small across the evaluated network series. This suggests that, under the considered spreading model, manipulating r does not substantially influence propagation effectiveness.

The reason for this may lie in the operational principles of the MICM and the corresponding geometric characteristics of the generated networks. Given that the effectiveness of spreading under this model depends primarily on local structures (such as degree or clustering coefficient) networks with similar values for these properties are expected to exhibit comparable diffusion dynamics. This appears to be the case in the considered example: manipulating r does not significantly affect either the degree distribution and the clustering coefficient (for reference, see the detailed analysis available in the GitHub repository). Consequently, the propagation proceeds in a similar manner across the evaluated networks.

Table 7

Change in Γ achieved through MICM-driven spreading in networks generated with **mABCD** with varying correlation strength between communities and the reference layer, relative to the results obtained on the baseline network (resembling *timik*).

δ	π	r_2, r_3				
		0.001	0.333	r_{limk}	0.667	1.000
AND	0.15	0.0007	0.0007	0.0000	0.0009	0.0015
	0.20	0.0016	0.0001	0.0000	0.0019	0.0022
	0.25	0.0010	-0.0003	0.0000	0.0003	0.0006
	0.30	0.0002	-0.0006	0.0000	-0.0004	-0.0002
OR	0.03	-0.0002	-0.0006	0.0000	-0.0010	-0.0011
	0.05	-0.0002	-0.0002	0.0000	-0.0004	-0.0002
	0.07	-0.0001	-0.0001	0.0000	-0.0002	-0.0002
	0.09	-0.0000	0.0001	0.0000	-0.0000	-0.0000

7. Conclusions

In this work, we have presented **mABCD**, a framework for generating multilayer networks with community structure and inter-layer relationships. We hope it addresses the existing gap in the field through its highly customizable configuration options and efficient algorithmic design. Furthermore, we release its implementation in Julia, along with Python ports, to make **mABCD** accessible to the broader research community. As **mABCD** builds upon the well-established generator introduced by Kamiński et al. (2021a), it can be regarded as a new member of the **ABCD** family. In addition to outlining the essential components of the model, we have validated its properties, demonstrating that **mABCD** produces stable results and outperforms its closest competitors in both efficiency and usage flexibility.

The paper also shows potential applications of the framework, illustrated in the context of studying spreading phenomena. Through three experiments, we examined whether and how selected parameters of the **mABCD** configuration model contribute to diffusion effectiveness. The results are nuanced rather than clear-cut. While there is a consistent trend with respect to the number of actors, the level of inter-community noise affects diffusion depending on the imposed regime. Finally, the strength of correlation between groups appears to have little influence on information spread. Altogether, these findings suggest that the proposed model is capable of supporting the analysis of spreading processes in multilayer networks, albeit up to certain extent.

As for the limitations of the proposed approach, the most significant one arises from the design assumptions underlying the model's development. Specifically, **mABCD** is capable of generating scale-free networks, and therefore its applications are restricted to systems that conform to this property (e.g., social ones). Another constraint concerns the size of the generated network. Although it should not be regarded as the major shortcoming, it is important to note that **mABCD** fails to properly generate structures when certain parameters are assigned excessively low values, such as the minimum degree (δ_i), the minimum community size (s_i), or the number of actors (n). The next limitation, which became particularly apparent during the experiments described in Section 6, concerns the difficulty of representing the properties of a real-world network within the parameter space of **mABCD**, especially the correlation strength between communities and the latent reference layer (r_i) which constitutes a core of the third phase in the network construction process (see Section 4.3). Although we have proposed a methodology for the coarse estimation of this value, a follow-up study is required to address this issue comprehensively.

Apart from the above, future work may proceed in two directions. The first concerns the **ABCD** model itself, which could be extended to other types of networks, such as temporal ones or multilayer ones without a lower bound imposed on the network size. The second way relates to the experiments involving spreading phenomena. In this context, it would be valuable to evaluate how other parameters of **mABCD** influ-

ence the dynamics of spreading processes. However, a more important and promising direction is the development of a metric for quantifying the difference between a real-world network and its “digital twin” generated by **mABCD**. Such a method could open a new avenue for influence maximization studies, as it would allow results and conclusions to be more directly related to real-world systems.

CRediT authorship contribution statement

Łukasz Kraiński: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization; **Michał Czuba**: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization; **Piotr Bródka**: Conceptualization, Methodology, Formal analysis, Investigation, Resources, Writing – original draft, Writing – review & editing, Supervision, Funding acquisition; **Paweł Prałat**: Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft, Writing – review & editing; **Bogumił Kamiński**: Conceptualization, Methodology, Software, Writing – review & editing, Supervision, Funding acquisition; **François Théberge**: Conceptualization, Writing – original draft, Writing – review & editing.

Ethics approval

This study did not involve any experiments on humans or animals, and therefore did not require ethical approval, clinical trial registration, or patient consent.

Materials from other sources

This study does not contain any material reproduced from other sources that required permission.

Data availability

The datasets generated during and/or analyzed during the current study and source code are available in following GitHub repositories: <https://github.com/KraińskiL/MLNABCDGraphGenerator.jl>, <https://github.com/anty-filidor/spreading-vs-mln-structure>. Access to the DVC storage will be granted upon reasonable request.

Funding

This research was partially supported by the National Science Centre, Poland, under Grant no. 2022/45/B/ST6/04145,⁹ the Polish Ministry of Education and Science within the programme “International Projects Co-Funded”, and the EU under the Horizon Europe, grant no. 101086321 OMINO. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the National Science Centre, Polish Ministry of Education and Science, the EU or the European Research Executive Agency.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Piotr Bródka reports financial support was provided by National Science Centre Poland. Piotr Bródka reports financial support was provided by European Union. Piotr Bródka reports financial support was provided by Republic of Poland Ministry of Science and Higher Education. Michał Czuba reports financial support was provided by National Science Centre Poland. Michał Czuba reports financial support was provided by European Union. Michał Czuba reports financial support was provided by

⁹ <https://multispread.pwr.edu.pl/>

Republic of Poland Ministry of Science and Higher Education. The other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- Alstott, J., Bullmore, E., & Plenz, D., (2014). Powerlaw: A python package for analysis of heavy-tailed distributions. *PloS One*, 9(1), 1–11. <https://doi.org/10.1371/journal.pone.0085777>
- Aref, S., Chheda, H., & Mostajabadeh, M. (2022). The Bayan algorithm: Detecting communities in networks through exact and approximate optimization of modularity. *arXiv preprint arXiv:2209.04562*
- Barabási, A.-L. (2016). *Network science*. Cambridge University Press. <http://barabasi.com/networksciencebook/>.
- Barrett, J., DeWolfe, R., Kamiński, B., Prałat, P., Smith, A., & Thérberge, F. (2025a). The artificial benchmark for community detection with outliers and overlapping communities (ABCD+o2). In *International workshop on modelling and mining networks* (pp. 125–140). Springer.
- Barrett, J., Kamiński, B., Prałat, P., & Thérberge, F. (2025b). Self-similarity of communities of the ABCD model. *Theoretical Computer Science*, 1026, 115012.
- Bazzi, M., Jeub, L. G. S., Arenas, A., Howison, S. D., & Porter, M. A. (2020). A framework for the construction of generative models for mesoscale structure in multilayer networks. *Physical Review Research*, 2(2), 023100.
- Bender, E. A., & Canfield, E. R. (1978). The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3), 296–307.
- Berlingerio, M., Coscia, M., & Giannotti, F. (2011). Finding and characterizing communities in multidimensional networks. In *2011 international conference on advances in social networks analysis and mining* (pp. 490–494). IEEE.
- Blondel, V. D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), P10008.
- Bollobás, B. (1980). A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal of Combinatorics*, 1(4), 311–316.
- Bródka, P. (2016). A method for group extraction and analysis in multilayer social networks. *arXiv preprint arXiv:1612.02377*
- Bródka, P., Chmiel, A., Magnani, M., & Ragozini, G. (2018). Quantifying layer similarity in multiplex networks: A systematic study. *Royal Society Open Science*, 5(8), 171747.
- Bródka, P., Czuba, M., Kamiński, B., Kraiński, Ł., Prałat, P., & Thérberge, F. (2025). The multilayer artificial benchmark for community detection (mABCD). In *International workshop on modelling and mining networks* (pp. 172–188). Springer.
- Bródka, P., Filipowski, T., & Kazienko, P. (2013). An introduction to community detection in multi-layered social network. In *Information systems, e-learning, and knowledge management research: 4th world summit on the knowledge society, WSKS 2011, Mykonos, Greece, September 21–23, 2011. Revised selected papers 4* (pp. 185–190). Springer.
- Cardillo, A., Gómez-Gardeñes, J., Zanin, M., Romance, M., Papo, D., Pozo, F. d., & Boccaletti, S. (2013). Emergence of network features from multiplexity. *Scientific Reports*, 3(1344), 1–6. <https://doi.org/10.1038/srep01344>
- Chen, W., Wang, Y., & Yang, S. (2009). Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '09* (p. 199–208). New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/1557019.1557047>
- Coleman, J., Katz, E., & Menzel, H. (1957). The diffusion of an innovation among physicians. *Sociometry*, 20(4), 253–270. <http://www.jstor.org/stable/2785979>
- Czuba, M., & Bródka, P. (2025). Rank-refining seed selection methods for budget constrained influence maximisation in multilayer networks under linear threshold model. *Social Network Analysis and Mining*, 15(1), 46. <https://doi.org/10.1007/s13278-025-01454-7>
- Czuba, M., Nurek, M., Serwata, D., Qi, Y.-X., Jia, M., Musial, K., Michalski, R., & Bródka, P. (2024). Network diffusion – framework to simulate spreading processes in complex networks. *Big Data Mining And Analytics*, (pp. 1–13). <https://doi.org/10.26599/BDMA.2024.9020010>
- De Domenico, M., Lancichinetti, A., Arenas, A., & Rosvall, M. (2015). Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Physical Review X*, 5, 011027. <https://doi.org/10.1103/PhysRevX.5.011027>
- De Luca, M., Fasolino, A. R., Ferraro, A., Moscato, V., Sperli, G., & Tramontana, P. (2023). A community detection approach based on network representation learning for repository mining. *Expert Systems with Applications*, 231, 120597. <https://doi.org/10.1016/j.eswa.2023.120597>
- Dickson, M. E., Magnani, M., & Rossi, L. (2016). *Multilayer social networks*. Cambridge, England, UK: Cambridge University Press. <https://doi.org/10.1017/CBO9781139941907>
- Eldardi, H., & Neville, J. (2010). Multi-network fusion for collective inference. In *Proceedings of the eighth workshop on mining and learning with graphs, MLG '10* (p. 46–54). New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/1830252.1830259>
- Elsis, A., Mandviwalla, A., Szymanski, B. K., & Sharkey, T. (2022). A network generator for covert network structures. *Information sciences*, 584, 387–398. <https://doi.org/10.1016/j.ins.2021.10.066>
- Erdős, P., Rényi, A. (1960). On the evolution of random graphs. *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, 5(1), 17–60. <https://static.renyi.hu/~p-erdos/1960-10.pdf>.
- Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3–5), 75–174.
- Fortunato, S., & Barthelemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1), 36–41.
- Goldenberg, J., Libai, B., & Muller, E. (2001). Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, 12(3), 211–223. <https://doi.org/10.1023/A:1011122126881>
- Gottfried, J. (2024). Americans' social media use. *Pew Research Center*, 31, 1–29.
- Granovetter, M. (1978). Threshold models of collective behavior. *American Journal of Sociology*, 83(6), 1420–1443. <http://www.jstor.org/stable/2778111>
- Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th Python in science conference* (pp. 11–15). Pasadena, CA USA. <https://aric.hagberg.org/papers/hagberg-2008-exploring.pdf>.
- Holland, P. W., Laskey, K. B., & Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks*, 5(2), 109–137.
- Huang, Y., Panahi, A., Krim, H., & Dai, L. (2019). Community detection and improved detectability in multiplex networks. *IEEE Transactions on Network Science and Engineering*, 7(3), 1697–1709.
- Jankowski, J., Michalski, R., & Bródka, P. (2017). A multilayer network dataset of interaction and influence spreading in a virtual world. *Scientific Data*, 4(1), 170144. <https://doi.org/10.1038/sdata.2017.144>
- Kamiński, B., Pankratz, B., Prałat, P., & Thérberge, F. (2022). Modularity of the ABCD random graph model with community structure. *Journal of Complex Networks*, 10(6), cnac050.
- Kamiński, B., Prałat, P., & Thérberge, F. (2021a). Artificial benchmark for community detection (ABCD)—Fast random graph model with community structure. *Network Science*, (pp. 1–26).
- Kamiński, B., Prałat, P., & Thérberge, F. (2021b). Mining complex networks. Chapman and Hall/CRC. <https://doi.org/10.1201/9781003218869>
- Kamiński, B., Prałat, P., & Thérberge, F. (2023a). Artificial benchmark for community detection with outliers (ABCD+o). *Applied Network Science*, 8(1), 25.
- Kamiński, B., Prałat, P., & Thérberge, F. (2023b). Hypergraph artificial benchmark for community detection (h-ABCD). *Journal of Complex Networks*, 11(4), cnad028.
- Kamiński, B., Olczak, T., Pankratz, B., Prałat, P., & Thérberge, F. (2022). Properties and performance of the ABCDe random graph model with community structure. *Big Data Research*, 30, 100348.
- Karimi, F., Lotfi, S., & Izadkhan, H. (2020). Multiplex community detection in complex networks using an evolutionary approach. *Expert Systems with Applications*, 146, 113184. <https://doi.org/10.1016/j.eswa.2020.113184>
- Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, 30(1–2), 81–93.
- Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., & Porter, M. A. (2014). Multilayer networks. *Journal of Complex Networks*, 2(3), 203–271. <https://doi.org/10.1093/comnet/cnu016>
- Lancichinetti, A., & Fortunato, S. (2009). Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1), 016118.
- Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4), 046110.
- Magnani, M., Hanteer, O., Interdonato, R., Rossi, L., & Tagarelli, A. (2021a). Community detection in multiplex networks. *ACM Computing Surveys (CSUR)*, 54(3), 1–35.
- Magnani, M., Rossi, L., & Vega, D. (2021b). Analysis of multiplex social networks with r. *Journal of Statistical Software*, 98, 1–30.
- Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., & Onnela, J.-P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 328(5980), 876–878. <https://doi.org/10.1126/science.1184819>
- Omodei, E., De Domenico, M., & Arenas, A. (2015). Characterizing interactions in online social networks during exceptional events. *Frontiers in Physics*, 3, 59.
- Orman, G. K., & Labatut, V. (2009). A comparison of community detection algorithms on artificial networks. In *Discovery science: 12th international conference, DS 2009, Porto, Portugal, October 3–5, 2009* (pp. 242–256). Springer.
- Pamfil, A. R., Howison, S. D., Lambiotte, R., & Porter, M. A. (2019). Relating modularity maximization and stochastic block models in multilayer networks. *SIAM Journal on Mathematics of Data Science*, 1(4), 667–698.
- Panayiotou, G., Wohler, I. K., Bask, M., Bask, M., Magnani, M., & Mäkinen, I. H. (2025). Anatomy of a Swedish population-scale network. *Scientific Reports*, 15(30300), 1–13. <https://doi.org/10.1038/s41598-025-15966-x>
- Paradowski, M. B., Whitby, N., Czuba, M., & Bródka, P. (2024). Peer interaction dynamics and second language learning trajectories during study abroad: A longitudinal investigation using dynamic computational social network analysis. *Language Learning*, 74(S2), 58–115.
- Popiel, A., Kazienko, P., & Kajdanowicz, T. (2015). MuNeG—The framework for multilayer network generator. In *2015 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 1316–1323). <https://doi.org/10.1145/2808797.2808902>
- Qing, H., & Wang, J. (2024). Bipartite mixed membership distribution-free model. A novel model for community detection in overlapping bipartite weighted networks. *Expert Systems with Applications*, 235, 121088. <https://doi.org/10.1016/j.eswa.2023.121088>
- Rossi, L., & Magnani, M. (2015). Towards effective visual analytics on multiplex and multilayer networks. *Chaos, Solitons & Fractals*, 72, 68–76. Multiplex Networks: Structure, Dynamics and Applications <https://doi.org/10.1016/j.chaos.2014.12.022>
- Snijders, T. A. B., Pattison, P. E., Robins, G. L., & Handcock, M. S. (2006). New specifications for exponential random graph models. *Sociological Methodology*, 36(1), 99–153. <https://doi.org/10.1111/j.1467-9531.2006.00176.x>
- Sun, Q., Chen, Z., Yang, B., Ji, C., Fu, X., Zhou, S., Peng, H., Li, J., & Yu, P. S. (2024). Ge-bench: An open and unified benchmark for graph condensation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, & C. Zhang (Eds.),

- Advances in neural information processing systems* (pp. 37900–37927). Curran Associates, Inc. (vol. 37). https://proceedings.neurips.cc/paper_files/paper/2024/file/42ed003e72eab5136e3d2fdbef4b8d7b-Paper-Datasets_and_Benchmarks_Track.pdf.
- Tang, L., Wang, X., & Liu, H. (2012). Community detection via heterogeneous interaction analysis. *Data Mining and Knowledge Discovery*, 25, 1–33.
- Tarvid, A. (2017). Stimulation model—a multilayer social network generator. In *2017 the European modeling and simulation symposium* (pp. 433–438). https://www.msc-les.org/proceedings/emss/emss2017/emss2017_433.html.
- Vinh, N. X., Epps, J., & Bailey, J. (2009). Information theoretic measures for clusterings comparison: Is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning* (pp. 1073–1080).
- Wątroba, P., & Bródka, P. (2023). Influence of information blocking on the spread of virus in multilayer networks. *Entropy*, 25(2), 231. <https://doi.org/10.3390/e25020231>
- Wormald, N. C. (1984). Generating random regular graphs. *Journal of Algorithms*, 5(2), 247–280.
- Wormald, N. C., Lamb, J. D., Preece, D. A. (1999). Models of random regular graphs. *London Mathematical Society Lecture Note Series*, (pp. 239–298).
- Zhong, Y. D., Srivastava, V., & Leonard, N. E. (2022). Influence spread in the heterogeneous multiplex linear threshold model. *IEEE Transactions on Control of Network Systems*, 9(3), 1080–1091. <https://doi.org/10.1109/TCNS.2021.3088782>

Appendix A

Scientific Achievements

The list of scientific achievements that were reached during the period of the doctoral education covers both those strictly related to this dissertation (Sec. A.1, A.2, A.3, A.4, A.5) and other activities contributing to the domain of Information and Communication Technology and the academia in general (Sec. A.6).

A.1 Research Projects

The research covered by this dissertation has been conducted as a part of the following scientific projects:

1. *Control and Spreading Processes in Multilayer Networks*, founded by the National Science Centre (Poland), project no. OPUS 2022/45/B/ST6/04145, 04.2023-07.2026;
2. *OMINO - Overcoming Multilevel INFORMATION Overload*, founded by the Horizon Europe (European Union), project number 101086321, 11.2024-02.2025.

A.2 Collaborators and Advisors

Publications comprising this dissertation are the outcome of collaboration with the following academics⁵:

1. Piotr Bielak, PhD — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;
2. Piotr Bródka, DSc PhD — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;

⁵Due to the lack of a one-to-one correspondence between Anglo-Saxon and Central European systems, the author has made a best-effort attempt to translate academic degrees (rather than positions) into their English equivalents. Any inaccuracies are sincerely apologised for.

3. Mingshan Jia, PhD — *Data Science Institute, University of Technology Sydney*, Sydney, Australia;
4. Prof. Bogumił Kamiński, DSc PhD — *Decision Analysis and Support Unit, SGH Warsaw School of Economics*, Warsaw, Poland;
5. Łukasz Kraiński, MA — *Decision Analysis and Support Unit, SGH Warsaw School of Economics*, Warsaw, Poland;
6. Radosław Michalski, DSc PhD — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;
7. Katarzyna Musiał-Gabryś, PhD — *Data Science Institute, University of Technology Sydney*, Sydney, Australia;
8. Mateusz Nurek, MSc — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;
9. Michał Paradowski, DSc PhD — *Institute of Applied Linguistics, University of Warsaw*, Warsaw, Poland;
10. Adam Piróg, MSc — *4Semantics Sp. z o.o.*, Białystok, Poland;
11. Paweł Prałat, PhD — *Department of Mathematics, Toronto Metropolitan University*, Toronto, Canada;
12. Yu-Xuan Qi, PhD — *Data Science Institute, University of Technology Sydney*, Sydney, Australia;
13. Damian Serwata, MSc — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;
14. Mateusz Stolarski, MSc — *Department of Artificial Intelligence, Wrocław University of Science and Technology*, Wrocław, Poland;
15. François Théberge, PhD — *Tutte Institute for Mathematics and Computing*, Ottawa, Canada;
16. Nicole Whitby, MA — *Brigham Young University*, Provo, USA.

A.3 Research Internships

My doctoral research involved visits and appointments to the following institutions:

1. *University Technology Sydney*, Faculty of Engineering and Information Technology; Sydney, Australia, 11.2024-02.2025, Visiting Scholar.
2. *University Technology Sydney*, Faculty of Engineering and Information Technology; Sydney, Australia, 01-02.2026, Visiting Scholar.

A.4 Scientific Conferences

During the doctoral education the author have actively participated in the following conferences whose scope covers the research domain considered here:

No.	Description	Presentation (oral / poster)
1.	Michał Czuba, Piotr Bródka; <i>Network Diffusion — a package for simulating spreading of multiple interacting processes</i> ; DSAA (9th International Conference on Data Science and Advanced Analytics); Shenzhen, China; 13–16.10.2022	oral (virtual)
2.	Michał Czuba, Piotr Bródka; <i>Seed Selection for Linear Threshold Model in Multilayer Networks</i> ; International Network for Social Network Analysis - Sunbelt 2023; Portland, USA; 27.06–01.07.2023	oral
3.	Michał Czuba, Piotr Bródka; <i>Seed Selection for Linear Threshold Model in Multilayer Networks</i> ; NetSci (International School and Conference on Network Science); Vienna, Austria, 10–14.07.2023	poster
4.	Michał Czuba, Mateusz Nurek, Damian Serwata, Yu-Xuan Qi, Mingshan Jia, Katarzyna Musiał, Radosław Michalski, Piotr Bródka; <i>Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks</i> ; WAW (19th Workshop on Modelling and Mining Networks); Warsaw, Poland; 03–06.06.2024	oral
5.	Michał Czuba, Mateusz Nurek, Damian Serwata, Yu-Xuan Qi, Mingshan Jia, Katarzyna Musiał, Radosław Michalski, Piotr Bródka; <i>Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks</i> ; NetSci (International School and Conference on Network Science); Quebec, Canada, 16–21.06.2024	poster
6.	Michał Czuba, Mateusz Nurek, Damian Serwata, Yu-Xuan Qi, Mingshan Jia, Katarzyna Musiał, Radosław Michalski, Piotr Bródka; <i>Network Diffusion — Framework to Simulate Spreading Processes in Complex Networks</i> ; IC2S2 (International Conference on Computational Social Science); Philadelphia, USA; 17–20.07.2024	oral
7.	Michał B. Paradowski, Piotr Bródka, Michał Czuba, Nicole Whitby; <i>Dynamic computational social network analysis of language learner interactions: Novel insights for study-abroad second language acquisition</i> ; SCiL (7th Meeting of Society for Computation in Linguistics); Irvine, USA; 27–29 2024	poster (virtual)
8.	Michał Czuba, Mateusz Stolarski, Adam Piróg, Piotr Bielak, Piotr Bródka; <i>Identifying Super Spreaders in Multilayer Networks</i> ; NetSci (International School and Conference on Network Science); Maastricht, Netherlands; 02–06.06.2025	poster (award)

9.	Piotr Bródka, Michał Czuba, Łukasz Kraiński, Bogumił Kamiński, Paweł Prałat, François Théberge; <i>The Multilayer Artificial Benchmark for Community Detection (mABCD)</i> ; WAW (20th Workshop on Modelling and Mining Networks); Vilnius, Lithuania; 30.06–03.07.2025	oral
10.	Michał Czuba, Mateusz Stolarski, Adam Piróg, Piotr Bielak, Piotr Bródka; <i>Identifying Super Spreaders in Multilayer Networks</i> ; Polskie Sympozjum Sztucznej Inteligencji; Wrocław, Poland; 11–12.09.2025	oral
11.	Michał Czuba, Mateusz Stolarski, Adam Piróg, Piotr Bielak, Piotr Bródka; <i>Identifying Super Spreaders in Multilayer Networks</i> ; ECAI (European Conference on Artificial Intelligence); Bologna, Italy; 25–30.10.2025	oral
12.	Łukasz Kraiński, Michał Czuba, Piotr Bródka, Paweł Prałat, Bogumił Kamiński, François Théberge; <i>Multilayer Artificial Benchmark for Community Detection (mABCD)</i> ; NetSciX (International School and Conference on Network Science); Auckland, New Zealand; 17–20.02.2026	poster
13.	Michał Czuba, Mingshan Jia, Piotr Bródka, Katarzyna Musiał; <i>Applicability of the Minimal Dominating Set for Influence Maximisation in Multilayer Networks</i> ; NetSciX (International School and Conference on Network Science); Auckland, New Zealand; 17–20.02.2026	oral

A.5 Journal and Conference Reviews

I have served as a reviewer in the following venues:

1. Social Network Analysis and Mining, **100 MNiSW points**, 2023, reviewed one article;
2. Network Science, **140 MNiSW points**, 2024, reviewed one article;
3. European Conference of Artificial Intelligence, **140 MNiSW points**, 2025, reviewed one article;
4. Intelligent Data Analysis, **70 MNiSW points**, 2025, reviewed one article.

A.6 Other Scientific Activities

During the period of doctoral studies, the author was engaged in research activities beyond a scope of the research problem considered within this dissertation. The most important achievements and ventures are listed below.

Co-authorship of papers

- P. Kawa et al., ‘*Improved DeepFake Detection Using Whisper Features*,’ in: Proc. INTERSPEECH 2023, pp. 4009–4013, 2023, DOI: 10.21437/Interspeech.2023-1537. **140 MNiSW points, core A**;
- K. Dykiert, M. Stolarski, W. Cieżak, M. Czuba and P. Bródka, ‘*Implementing Network Science to enhance water distribution network pipe roughness calibration*,’ in: CCWI 2025 - 21st Computing & Control for the Water Industry Conference, pp. 1–8, Aug., 2025, DOI: 10.15131/shef.data.29920964.v1. **20 MNiSW points**;
- P. Bródka, M. Czuba, B. Kamiński, Ł. Kraiński, P. Prałat and F. Théberge, ‘*The Multilayer Artificial Benchmark for Community Detection (mABCD)*,’ in: Modelling and Mining Networks, pp. 172–188, 2025, DOI: 10.1007/978-3-031-92898-7_12. **20 MNiSW points**;
- M. B. Paradowski, P. Bródka, M. Czuba and N. Whitby, ‘*Dynamic computational social network analysis of language learner interactions: Novel insights for study-abroad second language acquisition*,’ in: Society for Computation in Linguistics, vol. 7, 1, Jun., 2024, DOI: 10.7275/sci1.2218. **20 MNiSW points**.

The last two publications mentioned above present preliminary research that was included in this dissertation respectively as **P-7** and **P-3**, therefore they are mentioned merely here.

Bibliometrics

According to the Google Scholar database, the author’s complete body of work has accrued the metrics as presented in Tab. A.2.

Table A.2: Bibliometric scores for all works by the author, as of 4th March 2026.

Metric	Value
Citations	124
h-index	4
i10-index	2

Involvement in self-governing bodies of WUST

- Participation in the Council of the Faculty of Information and Communication Technology as a deputy representing PhD students; Term 2020-2024, 2024-2028;
- Participation in the Council for the Teaching Quality at the Faculty of Information and Communication Technology as a deputy representing PhD students; Term 2024-2028.

Additional activities

- Preparation of educational materials for the course "System and Decision Methods" for undergraduate students of Applied Computer Science at WUST, Wrocław, Poland, 2022;
- Preparation of educational materials for the course "Complex Networks" for Master's students of Applied Computer Science at WUST, Wrocław, Poland, 2023.
- Participation in the AI-Tech project (<https://www.gov.pl/web/aitech>) through the preparation of educational materials and delivery of courses concerning aspects of network science during the summer school, Warsaw, Poland, 05–06.2023;
- Participation in the TUD|excite Summer School "Patents in Search of New Business", organised by Technische Universität Dresden (recipient of the Best Project Award), Hoyerswerda, Germany, 25–31.08.2024;
- Awardee of the Scientific Scholarship Programme of the Municipality of Wrocław, Wrocław, Poland, 2025.

Participation in research and development projects

- Vestigit Watermark on the Edge — Platform for Securing Streaming in Multi-CDN Environments Using Content Adaptive Fast Watermark (CAFW), project no.: POIR.01.01.01-00-0090/22, sources of funding: National Centre for Research and Development, Vestigit sp. z o.o., implementation period: 01-12.2023, position: Data Scientist;
- Adspective — AI-Powered Automation Platform for High-Quality Multimedia Advertisement Creation, project no.: POIR.01.01.01-00-0218/21, sources of funding: National Centre for Research and Development, Vestigit sp. z o.o., implementation period: 01.2022-09.2023, position: Senior Developer.

Bibliography

- [1] A.-L. Barabási and M. Pósfai, *Network science*, 2016, [Online]. Available: <http://barabasi.com/networksciencebook/>.
- [2] J. Ma et al., ‘*Computational Social Science for Nonprofit Studies: Developing a Toolbox and Knowledge Base for the Field*,’ in: *Voluntas*, vol. 34, no. 1, pp. 52–63, Feb., 2023, DOI: 10.1007/s11266-021-00414-x.
- [3] H. Hexmoor, *Computational Network Science: An Algorithmic Approach*, 2016, DOI: 10.1016/C2013-0-19272-0.
- [4] Y. Gao et al., ‘*Hypergraph Learning: Methods and Practices*,’ in: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2548–2566, 2022, DOI: 10.1109/TPAMI.2020.3039374.
- [5] C. Shi, X. Wang and P. S. Yu, *Heterogeneous Graph Representation Learning and Applications*, 2022, DOI: 10.1007/978-981-16-6166-2_1.
- [6] P. Holme and J. Saramäki, ‘*Temporal networks*,’ in: *Physics Reports*, vol. 519, no. 3, pp. 97–125, 2012, DOI: <https://doi.org/10.1016/j.physrep.2012.03.001>.
- [7] H. Dong and K. Wang, ‘*Interaction event network modeling based on temporal point process*,’ in: *IJSE Transactions*, vol. 54, no. 7, pp. 630–642, 2022, DOI: 10.1080/24725854.2021.1906468.
- [8] M. Kivelä et al., ‘*Multilayer Networks*,’ in: *Journal of complex networks*, vol. 2, no. 3, pp. 203–271, 2014, DOI: 10.1093/comnet/cnu016.
- [9] J. Jankowski, R. Michalski and P. Bródka, ‘*A multilayer network dataset of interaction and influence spreading in a virtual world*,’ in: *Scientific Data*, vol. 4, no. 1, p. 170144, 2017, DOI: 10.1038/sdata.2017.144.
- [10] A. Cardillo et al., ‘*Emergence of network features from multiplexity*,’ in: *Scientific Reports*, vol. 3, no. 1344, pp. 1–6, Feb., 2013, DOI: 10.1038/srep01344.
- [11] J. D. Wilson et al., ‘*Analysis of population functional connectivity data via multilayer network embeddings*,’ in: *Network Science*, vol. 9, no. 1, pp. 99–122, 2021, DOI: 10.1017/nws.2020.39.
- [12] N. Musmeci et al., ‘*The Multiplex Dependency Structure of Financial Markets*,’ in: *Complexity*, vol. 2017, no. 1, p. 9586064, Jan., 2017, DOI: 10.1155/2017/9586064.

- [13] G. Panayiotou, M. Magnani and B. Pinaud, '*Current challenges in multilayer network engineering*,' in: Applied Network Science, vol. 9, no. 1, pp. 1–23, Dec., 2024, DOI: 10.1007/s41109-024-00686-4.
- [14] A. Aleta et al., '*Multilayer network science: theory, methods, and applications*,' in: arXiv preprint, 2026, DOI: 10.48550/arXiv.2511.23371.
- [15] K. W. Ogilvy and G. McKendrick A., '*A contribution to the mathematical theory of epidemics*,' in: Proc. R. Soc. London A, vol. 115, no. 772, pp. 700–721, Aug., 1927, DOI: 10.1098/rspa.1927.0118.
- [16] Disease Outbreak News, [Online; accessed 29. Oct. 2025], Oct. 2025. [Online]. Available: <https://www.who.int/emergencies/disease-outbreak-news>.
- [17] M. E. J. Newman, '*Spread of epidemic disease on networks*,' in: Physical Review E, vol. 66, p. 016128, Jul., 2002, DOI: 10.1103/PhysRevE.66.016128.
- [18] C. Nowzari, V. M. Preciado and G. J. Pappas, '*Analysis and Control of Epidemics: A Survey of Spreading Processes on Complex Networks*,' in: IEEE Control Systems Magazine, vol. 36, no. 1, pp. 26–46, 2016, DOI: 10.1109/MCS.2015.2495000.
- [19] Z. Wang et al., '*Coupled disease-behavior dynamics on complex networks: A review*,' in: Physics of Life Reviews, vol. 15, pp. 1–29, 2015, DOI: <https://doi.org/10.1016/j.plrev.2015.07.006>.
- [20] D. J. Watts and P. S. Dodds, '*Influentials, Networks, and Public Opinion Formation*,' in: Journal of Consumer Research, vol. 34, no. 4, pp. 441–458, May, 2007, DOI: 10.1086/518527.
- [21] B. Doerr, M. Fouz and T. Friedrich, '*Why rumors spread so quickly in social networks*,' in: Communications of the ACM, vol. 55, no. 6, pp. 70–75, Jun., 2012, DOI: 10.1145/2184319.2184338.
- [22] E. Aïmeur, S. Amri and G. Brassard, '*Fake news, disinformation and misinformation in social media: a review*,' in: Social Network Analysis and Mining, vol. 13, no. 1, pp. 30–36, Feb., 2023, DOI: 10.1007/s13278-023-01028-5.
- [23] J. Mandravickaitė et al., '*Analyzing and Identifying Russian Disinformation in Telegram Using Graph-Based Methods*,' in: 2025 International Conference on Military Communication and Information Systems (ICMCIS), pp. 1–9, 2025, DOI: 10.1109/ICMCIS64378.2025.11047922.
- [24] Y. Wang et al., '*Node centrality analysis of multiplex networks under Computer virus spreading*,' in: Proceedings of the 3rd International Conference on Robotics, Control and Automation, pp. 255–260, 2018, DOI: 10.1145/3265639.3265682.
- [25] P. Crucitti, V. Latora and M. Marchiori, '*Model for cascading failures in complex networks*,' in: Physical Review E, vol. 69, p. 045104, Apr., 2004, DOI: 10.1103/PhysRevE.69.045104.

- [26] P. Gai and S. Kapadia, '*Contagion in financial networks*,' in: Proceedings of the Royal Society A, vol. 466, no. 2120, pp. 2401–2423, Aug., 2010, DOI: 10.1098/rspa.2009.0410.
- [27] D. Lancour et al., '*One for all and all for One: Improving replication of genetic studies through network diffusion*,' in: PLoS Genet., vol. 14, no. 4, e1007306, Apr., 2018, DOI: 10.1371/journal.pgen.1007306.
- [28] D. Kempe, J. Kleinberg and É. Tardos, '*Maximizing the Spread of Influence Through a Social Network*,' in: 9th ACM SIGKDD international conference on Knowledge discovery and data mining, p. 137, 2003.
- [29] R. Pastor-Satorras and A. Vespignani, '*Immunization of complex networks*,' in: Physical Review E, vol. 65, p. 036 104, Apr., 2002, DOI: 10.1103/PhysRevE.65.036104.
- [30] Y.-Y. Liu and A.-L. Barabási, '*Control principles of complex systems*,' in: Reviews of Modern Physics, vol. 88, p. 035 006, 2016, DOI: 10.1103/RevModPhys.88.035006.
- [31] J. Leskovec et al., '*Cost-effective outbreak detection in networks*,' in: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 420–429, 2007, DOI: 10.1145/1281192.1281239.
- [32] A. Topîrceanu and M. Udrescu, '*Fast colonization algorithm for seed selection in complex networks based on community detection*,' in: Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 214–218, 2022, DOI: 10.1145/3487351.3488319.
- [33] Y. Ni, '*Sequential seeding to optimize influence diffusion in a social network*,' in: Applied Soft Computing, vol. 56, pp. 730–737, 2017, DOI: <https://doi.org/10.1016/j.asoc.2016.04.025>.
- [34] J. Jankowski et al., '*Balancing Speed and Coverage by Sequential Seeding in Complex Networks*,' in: Scientific Reports, vol. 7, no. 891, pp. 1–11, Apr., 2017, DOI: 10.1038/s41598-017-00937-8.
- [35] A. Zarezade et al., '*Steering Social Activity: A Stochastic Optimal Control Point Of View*,' in: Journal of Machine Learning Research, vol. 18, no. 205, pp. 1–35, 2018, [Online]. Available: <http://jmlr.org/papers/v18/17-416.html>.
- [36] C. Buono and L. A. Braunstein, '*Immunization strategy for epidemic spreading on multilayer networks*,' in: Europhysics Letters, vol. 109, no. 2, p. 26 001, Jan., 2015, DOI: 10.1209/0295-5075/109/26001.
- [37] C. Budak, D. Agrawal and A. El Abbadi, '*Limiting the spread of misinformation in social networks*,' in: Proceedings of the 20th International Conference on World Wide Web, pp. 665–674, 2011, DOI: 10.1145/1963405.1963499.
- [38] P. Bródka et al., '*Quantifying layer similarity in multiplex networks: a systematic study*,' in: R. Soc. Open Sci., vol. 5, no. 8, Aug., 2018, DOI: 10.1098/rsos.171747.

- [39] M. De Domenico, ‘*More is different in real-world multilayer networks*,’ in: Nature Physics, vol. 19, no. 9, pp. 1247–1262, Sep., 2023, DOI: 10.1038/s41567-023-02132-1.
- [40] J. Corneek et al., ‘*Simultaneous global and local clustering in multiplex networks with covariate information*,’ in: Journal of Complex Networks, vol. 14, no. 1, cnaf049, Jan., 2026, DOI: 10.1093/comnet/cnaf049.
- [41] P. Bródka, K. Musial and J. Jankowski, ‘*Interacting spreading processes in multilayer networks: a systematic review*,’ in: IEEE Access, vol. 8, pp. 10 316–10 341, 2020, DOI: 10.1109/ACCESS.2020.2965547.
- [42] Y. D. Zhong, V. Srivastava and N. E. Leonard, ‘*Influence Spread in the Heterogeneous Multiplex Linear Threshold Model*,’ in: IEEE Transactions on Control of Network Systems, vol. 9, no. 3, pp. 1080–1091, 2022, DOI: 10.1109/TCNS.2021.3088782.
- [43] K. Venkatakrishna Rao and C. R. Chowdary, ‘*CBIM: Community-based influence maximization in multilayer networks*,’ in: Information Sciences, vol. 609, pp. 578–594, 2022, DOI: 10.1016/j.ins.2022.07.103.
- [44] Y. Sun et al., ‘*Epidemic spreading on weighted complex networks*,’ in: Physics Letters A, vol. 378, no. 7, pp. 635–640, 2014, DOI: <https://doi.org/10.1016/j.physleta.2014.01.004>.
- [45] M. Granovetter, ‘*Threshold models of collective behavior*,’ in: American journal of sociology, vol. 83, no. 6, pp. 1420–1443, 1978, [Online]. Available: <http://www.jstor.org/stable/2778111>.
- [46] T. Harko, F. S. Lobo and M. Mak, ‘*Exact analytical solutions of the Susceptible-Infected-Recovered (SIR) epidemic model and of the SIR model with equal death and birth rates*,’ in: Applied Mathematics and Computation, vol. 236, pp. 184–194, 2014, DOI: <https://doi.org/10.1016/j.amc.2014.03.030>.
- [47] S. S. Singh et al., ‘*Influence maximization frameworks, performance, challenges and directions on social network: A theoretical study*,’ in: Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 9, pp. 7570–7603, 2022, DOI: 10.1016/j.jksuci.2021.08.009.
- [48] F. Erlandsson, P. Bródka and A. Borg, ‘*Seed selection for information cascade in multilayer networks*,’ in: Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications), pp. 426–436, 2018.
- [49] D. Zhao et al., ‘*Identifying influential spreaders in interconnected networks*,’ in: Physica Scripta, vol. 89, no. 1, p. 015 203, 2013.
- [50] M. Kitsak et al., ‘*Identification of influential spreaders in complex networks*,’ in: Nature Phys, vol. 6, pp. 888–893, Nov., 2010, DOI: 10.1038/nphys1746.

- [51] C. Ni and J. Yang, ‘Two-step seeding strategy in multiplex networks with inter-layer conversion cost of influence,’ in: Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 32, no. 8, p. 083 135, Aug., 2022, DOI: 10.1063/5.0096740.
- [52] H. Li et al., ‘PIANO: Influence Maximization Meets Deep Reinforcement Learning,’ in: IEEE Transactions on Computational Social Systems, vol. 10, no. 3, pp. 1288–1300, 2023, DOI: 10.1109/TCSS.2022.3164667.
- [53] A. A. Hagberg, D. A. Schult and P. J. Swart, ‘Exploring Network Structure, Dynamics, and Function Using NetworkX,’ in: Proceedings of the 7th Python in Science Conference, pp. 11–15, 2008.
- [54] G. Rossetti et al., ‘NDlib: A Python Library to Model and Analyze Diffusion Processes Over Complex Networks,’ in: International Journal of Data Science and Analysis, vol. 5, no. 1, pp. 61–79, Feb., 2018, DOI: 10.1007/s41060-017-0086-6.
- [55] M. Magnani, L. Rossi and D. Vega, ‘Analysis of Multiplex Social Networks with R,’ in: Journal of Statistical Software, vol. 98, no. 8, pp. 1–30, 2021, DOI: 10.18637/jss.v098.i08.
- [56] B. F. Maier and D. Brockmann, ‘Effective containment explains subexponential growth in recent confirmed COVID-19 cases in China,’ in: Science, vol. 368, no. 6492, pp. 742–746, 2020, DOI: 10.1126/science.abb4557.
- [57] A. Sadaf et al., ‘Maximising Influence Spread in Complex Networks by Utilising Community-Based Driver Nodes as Seeds,’ in: Information Management and Big Data, pp. 126–141, 2023, DOI: 10.1007/978-3-031-35445-8_10.
- [58] J. Coleman, E. Katz and H. Menzel, ‘The Diffusion of an Innovation Among Physicians,’ in: Sociometry, vol. 20, no. 4, pp. 253–270, 1957, [Online]. Available: <http://www.jstor.org/stable/2785979>.
- [59] L. Rossi and M. Magnani, ‘Towards effective visual analytics on multiplex and multilayer networks,’ in: Chaos, Solitons & Fractals, vol. 72, pp. 68–76, 2015, DOI: 10.1016/j.chaos.2014.12.022.
- [60] T. A. B. Snijders et al., ‘New Specifications for Exponential Random Graph Models,’ in: Sociological Methodology, vol. 36, no. 1, pp. 99–153, 2006, DOI: 10.1111/j.1467-9531.2006.00176.x.
- [61] Q. Lu, Z. Bu and Y. Wang, ‘A Multiobjective Evolutionary Approach for Influence Maximization in Multilayer Networks,’ in: Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence, pp. 431–438, 2020, DOI: 10.1145/3404555.3404568.
- [62] R. Michalski, P. Kazienko and J. Jankowski, ‘Convince a Dozen More and Succeed – The Influence in Multi-layered Social Networks,’ in: 2013 International Conference on Signal-Image Technology & Internet-Based Systems, pp. 499–505, 2013, DOI: 10.1109/SITIS.2013.85.

- [63] M. Magnani et al., ‘*Community Detection in Multiplex Networks*,’ in: ACM Computing Surveys, vol. 54, no. 3, May, 2021, DOI: 10.1145/3444688.
- [64] Y. Huang et al., ‘*Community Detection and Improved Detectability in Multiplex Networks*,’ in: IEEE Transactions on Network Science and Engineering, vol. 7, no. 3, pp. 1697–1709, 2020, DOI: 10.1109/TNSE.2019.2949036.
- [65] A. R. Pamfil et al., ‘*Relating modularity maximization and stochastic block models in multilayer networks*,’ in: SIAM Journal on Mathematics of Data Science, vol. 1, no. 4, pp. 667–698, 2019, DOI: 10.1137/18M1231304.
- [66] M. Bazzi et al., ‘*A framework for the construction of generative models for mesoscale structure in multilayer networks*,’ in: Phys. Rev. Res., vol. 2, p. 023100, Apr., 2020, DOI: 10.1103/PhysRevResearch.2.023100.
- [67] P. Bródka, J. Jankowski and R. Michalski, ‘*Sequential seeding in multilayer networks*,’ in: Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 31, no. 3, p. 033130, Mar., 2021, DOI: 10.1063/5.0023427.
- [68] V. R. K. and C. R. Chowdary, ‘*K++ Shell: Influence maximization in multilayer networks using community detection*,’ in: Computer Networks, vol. 234, p. 109916, 2023, DOI: 10.1016/j.comnet.2023.109916.
- [69] Y. Chen et al., ‘*Maximizing multiple influences and fair seed allocation on multilayer social networks*,’ in: PLOS ONE, vol. 15, no. 3, e0229201, Mar., 2020, DOI: 10.1371/journal.pone.0229201.
- [70] A. Kuhnle et al., ‘*Multiplex Influence Maximization in Online Social Networks With Heterogeneous Diffusion Models*,’ in: IEEE Transactions on Computational Social Systems, vol. 5, no. 2, pp. 418–429, 2018, DOI: 10.1109/TCSS.2018.2813262.
- [71] M. Newman, Networks: An Introduction, 2010, DOI: 10.1093/acprof:oso/9780199206650.001.0001.
- [72] P. Mitchell, Central Heating, Installation, Maintenance and Repair, 2008, [Online]. Available: <https://books.google.pl/books?id=S64mLfBs2vAC>.
- [73] Y.-Y. Liu, J.-J. Slotine and A.-L. Barabási, ‘*Controllability of complex networks*,’ in: Nature, vol. 473, no. 7346, pp. 167–173, 2011, DOI: 10.1038/nature10011.
- [74] M. Doostmohammadian, ‘*Minimal Driver Nodes for Structural Controllability of Large-Scale Dynamical Systems: Node Classification*,’ in: IEEE Systems Journal, vol. 14, no. 3, pp. 4209–4216, 2020, DOI: 10.1109/JSYST.2019.2956501.
- [75] J. C. Nacher and T. Akutsu, ‘*Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control*,’ in: New Journal of Physics, vol. 14, no. 7, p. 073005, 2012, DOI: 10.1088/1367-2630/14/7/073005.
- [76] J. C. Nacher et al., ‘*Finding and analysing the minimum set of driver nodes required to control multilayer networks*,’ in: Scientific Reports, vol. 9, no. 576, pp. 1–12, Jan., 2019, DOI: 10.1038/s41598-018-37046-z.

- [77] S. Tian et al., ‘*Deep Reinforcement Learning-Based Approach to Tackle Topic-Aware Influence Maximization*,’ in: Data Science and Engineering, vol. 5, no. 1, pp. 1–11, Mar., 2020, DOI: 10.1007/s41019-020-00117-1.
- [78] C. Ling et al., ‘*Deep graph representation learning and optimization for influence maximization*,’ in: Proceedings of the 40th International Conference on Machine Learning, 2023, DOI: 10.5555/3618408.3619288.
- [79] Z. Yuan, M. Shao and Z. Chen, ‘*Graph Bayesian Optimization for Multiplex Influence Maximization*,’ in: Proceedings of the AAAI Conference on Artificial Intelligence, Mar., 2024, DOI: 10.48550/arXiv.2403.18866.
- [80] Y. He et al., ‘*GNNRank: Learning Global Rankings from Pairwise Comparisons via Directed Graph Neural Networks*,’ in: Proceedings of the 39th International Conference on Machine Learning, vol. 162, pp. 8581–8612, Jul., 2022, [Online]. Available: <https://proceedings.mlr.press/v162/he22b.html>.
- [81] Q. Liu et al., ‘*Graph Neural Network for Tag Ranking in Tag-enhanced Video Recommendation*,’ in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 2613–2620, 2020, DOI: 10.1145/3340531.3416021.
- [82] B. Doerr, T. Friedrich and T. Sauerwald, ‘*Quasirandom Rumor Spreading: Expanders, Push vs. Pull, and Robustness*,’ in: Automata, Languages and Programming, pp. 366–377, 2009, DOI: 10.1007/978-3-642-02927-1_31.
- [83] R. Khallaf et al., ‘*Review of Digital Twins for Constructed Facilities*,’ in: Buildings, vol. 12, no. 11, 2022, DOI: 10.3390/buildings12112029.
- [84] M. Grieves, ‘*Digital Twin: Manufacturing Excellence through Virtual Factory Replication*,’ in: pp. 1–7, Mar., 2015, [Online]. Available: <https://orkg.org/papers/R154580>.
- [85] S. Kaewunruen et al., ‘*Digital Twins for Managing Railway Bridge Maintenance, Resilience, and Climate Change Adaptation*,’ in: Sensors, vol. 23, no. 1, 2023, DOI: 10.3390/s23010252.
- [86] robonation, ‘*RoboSub 2018: Final Standings - RoboSub*,’ in: RoboSub, Aug., 2021, [Online]. Available: <https://robosub.org/2018/08/05/2018-final-standings>.
- [87] J. Wen, B. Gabrys and K. Musial, ‘*Towards Digital Twin Oriented Modeling of Complex Networked Systems and Their Dynamics: A Comprehensive Survey*,’ in: IEEE Access, vol. 10, pp. 66 886–66 923, 2022, DOI: 10.1109/ACCESS.2022.3184801.
- [88] Y. Le Cun et al., ‘*Learning Algorithms For Classification: A Comparison On Hand-written Digit Recognition*,’ in: Neural Networks: The Statistical Mechanics Perspective, pp. 261–276, Aug., 1995, DOI: 10.1142/2808.
- [89] T. N. Kipf and M. Welling, ‘*Semi-Supervised Classification with Graph Convolutional Networks*,’ in: arXiv Preprint, Sep., 2016, DOI: 10.48550/arXiv.1609.02907.
- [90] M. Stolarski, A. Piróg and P. Bródka, ‘*Identifying Key Nodes for the Influence Spread Using a Machine Learning Approach*,’ in: Entropy, vol. 26, no. 11, 2024, DOI: 10.3390/e26110955.

- [91] M. De Domenico et al., ‘*Identifying Modular Flows on Multilayer Networks Reveals Highly Overlapping Organization in Interconnected Systems*,’ in: Phys. Rev. X, vol. 5, p. 011027, Mar., 2015, DOI: 10.1103/PhysRevX.5.011027.
- [92] P. Erdős, A. Rényi et al., ‘*On the evolution of random graphs*,’ in: Publ. math. inst. hung. acad. sci., vol. 5, no. 1, pp. 17–60, 1960, [Online]. Available: https://static.renyi.hu/~p_erdos/1960-10.pdf.
- [93] B. Bollobas et al., ‘*Directed Scale-Free Graphs*,’ in: Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 132–139, Jan., 2003, [Online]. Available: <https://www.microsoft.com/en-us/research/publication/directed-scale-free-graphs/>.
- [94] B. Kamiński et al., ‘*Modularity of the ABCD random graph model with community structure*,’ in: Journal of Complex Networks, vol. 10, no. 6, cnac050, 2022.
- [95] S. Carmi et al., ‘*A model of Internet topology using k -shell decomposition*,’ in: Proceedings of the National Academy of Sciences, vol. 104, no. 27, pp. 11 150–11 154, 2007, DOI: 10.1073/pnas.0701175104.
- [96] L. Page et al., ‘*The pagerank citation ranking: Bringing order to the web.*,’ Stanford InfoLab, Technical Report 1999-66, 1999, Previous number = SIDL-WP-1999-0120. [Online]. Available: <http://ilpubs.stanford.edu:8090/422/>.
- [97] J.-X. Zhang et al., ‘*Identifying a set of influential spreaders in complex networks*,’ in: Scientific reports, vol. 6, no. 1, p. 27 823, 2016, DOI: 10.1038/srep27823.
- [98] V. R. K, M. Katukuri and M. Jagarapu, ‘*CIM: clique-based heuristic for finding influential nodes in multilayer networks*,’ in: Applied Intelligence, vol. 52, no. 5, pp. 5173–5184, Mar., 2022, DOI: 10.1007/s10489-021-02656-0.
- [99] W. Chen, Y. Wang and S. Yang, ‘*Efficient influence maximization in social networks*,’ in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 199–208, 2009, DOI: 10.1145/1557019.1557047.
- [100] A. Casado et al., ‘*An iterated greedy algorithm for finding the minimum dominating set in graphs*,’ in: Mathematics and Computers in Simulation, vol. 207, pp. 41–58, 2023, DOI: 10.1016/j.matcom.2022.12.018.

AI disclosure. The use of large language models in this dissertation was limited to checking the orthography and grammar of the text, which was prepared exclusively by the author, for whom English is not a native language.