Pamukkale Univ Muh Bilim Derg, 31(2), 155-165, 2025

Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi

Pamukkale University Journal of Engineering Sciences

Optimizing influence propagation in directed networks: Novel formulations

Yönlendirilmiş ağlarda etki yayılımını eniyileme: Yeni formülasyonlar

Gökhan KARAKÖSE[1*]

[1]Department of Industrial Engineering, Faculty of Engineering, Architecture and Design, Bartin University, Bartın, Türkiye.
gkarakose@bartin.edu.tr

Received/Geliş Tarihi: 24.01.2024 Revision/Düzeltme Tarihi: 07.06.2024 doi: 10.5505/pajes.2024.19940
Accepted/Kabul Tarihi: 14.07.2024 Research Article/Araştırma Makalesi

Abstract

This paper aims to identify influential nodes in complex networks in a short period of time by proposing novel formulations. Traditional centrality metrics have ranked nodes based on individual centrality values, which fall short in identifying several influential nodes simultaneously. Recent literature has introduced an optimization model as a solution to this limitation; however, this model has some shortcomings such as long solution return time and high memory usage. In this paper, two novel formulations are presented as alternatives to this optimization model, with a primary goal of reducing the time needed to obtain solutions. Computational tests have shown that whereas the existing model is unable to return a solution within a 5-hour time frame for a small network with approximately 5,000 nodes, the proposed formulations can identify the most influential nodes within minutes, even for large networks with more than 100,000 nodes. The superiority of the proposed models actually lies in their significant reduction in the number of constraints and variables compared to the existing model. Additionally, this paper introduces a novel alternative formulation that addresses the overlapping effect observed in the previous formulations. Computational tests have shown that this model surpasses its predecessors in accelerating the spread of influence throughout the network without causing additional computational burden, thereby setting a better benchmark for future studies in this field.

Keywords: Influence maximization, Influential nodes, Optimization, Degree centrality, Mathematical modelling.

Öz

Bu makale, yeni formülasyonlar önererek karmaşık ağlardaki etkili düğümleri kısa zaman diliminde belirlemeyi amaçlamaktadır. Geleneksel merkeziyet ölçümleri, düğümleri bireysel merkeziyet değerlerine göre sıralamaktadır, bu da aynı anda birden fazla etkili düğümün belirlenmesinde yetersiz kalmaktadır. Güncel literatür bu kısıtlamaya çözüm olarak bir optimizasyon modeli sunmuştur, ancak bu modelin uzun süren çözüm döndürme süresi ve yüksek bellek kullanımı gibi bazı eksiklikleri vardır. Bu makalede, çözümleri elde etmede gereken süreyi azaltma ana amacıyla bu optimizasyon modeline alternatif olarak iki yeni formülasyon sunulmuştur. Hesaplamalı testler, mevcut modelin yaklaşık 5,000 düğümlü küçük bir ağ için 5 sa.'lik bir zaman dilimi içinde çözümü döndürmezken, önerilen formülasyonların 100,000'den fazla düğümlü büyük ağlar için bile en etkili düğümleri dakikalar içinde belirleyebildiğini göstermiştir. Önerilen modellerin üstünlüğü aslında mevcut modele kıyasla kısıtların ve değişkenlerin sayısının önemli ölçüde azaltılmasında yatmaktadır. Ek olarak, bu makale, önceki formülasyonlarda gözlenen örtüşen etki sorununu ele alan yeni bir alternatif formülasyon tanıtmaktadır. Hesaplamalı testler, bu modelin, ek hesaplama yüküne neden olmadan etki yayılımını ağ boyunca hızlandırmada öncekilerden daha üstün olduğunu, böylece bu alanda gelecekteki çalışmalar için daha iyi bir kıyaslama oluşturduğunu göstermiştir.

Anahtar kelimeler: Etki maksimizasyonu, Etkili düğümler, Optimizasyon, Derece merkezlilik, Matematiksel modelleme.

1 Introduction

Complex networks that abstractly represent the interactions in various real-world systems help us to understand the complexities of complex systems. Within these networks, individual elements are represented as nodes, while their connections are depicted as edges. Recent practical findings have triggered increasing interest in understanding the importance of nodes in complex networks in areas such as disease control [1],[2], marketing strategies [3],[4], the dynamics of public sentiment and rumors [5]-[8], epidemic modeling such as Covid-19 [9], spreading rumors [10],[11], dissemination of desired information [12], and protein-protein interactions [13].

The Influential Node Identification Problem (INIP) stands as a pivotal quest, aiming to identify a set of K influential nodes that exert maximum influence on others within these complex networks. For example, INIP was utilized for targeted advertising in marketing purposes in [14], as well as in public health initiatives to identify pivotal individuals for vaccination programs in [15],[16]. In [17], INIP was used to analyze significant countries (and regions) within the global economic system, whereas in [18], it was employed to identify influential nodes and critical lines within power transmission networks. In big data analysis, Szklarczyk et al. [19] applied INIP to uncover critical information within extensive datasets, while INIP was utilized to study essential neurons within brain neural networks in [20].

The literature addresses the INIP by utilizing various established centrality metrics. Freeman [21] examined Degree, Closeness, and Betweenness Centrality in human communication networks. The importance of a node was determined by its degree, betweenness, or closeness, depending on the context: degree measures communication activity, betweenness measures control over communication, and closeness measures independence or efficiency. Eigenvector Centrality and Alpha Centrality were proposed to measure the importance of nodes in a given network [22]. While Alpha Centrality can be applied to all networks, Eigenvector Centrality is applicable to networks where the status of nodes is influenced by other nodes they contact. These

*Corresponding author/Yazışılan Yazar

two methods are equivalent when both are applicable. Katz Centrality, which extends the concept of Eigenvector Centrality, measures the relative importance of a node based on the centrality of its neighbors and was first proposed by [23]. Later, an algorithm based on Katz Centrality was developed to solve the influence maximization problem [24]. Rehm et al. [25] utilized Katz Centrality to predict how a medical condition and its developments might impact astronaut productivity. Load Centrality, which slightly differs from Betweenness Centrality, was originally proposed by [26]. This distinction arises from the fact that Load Centrality uses a random walk technique to consider all paths rather than solely focusing on shortest paths like Betweenness Centrality. Harmonic Centrality, based on the sum of the inverses of the shortest path lengths, was introduced by [27]. This metric resolves the primary issue with Closeness Centrality, particularly the existence of pairs of nodes that are unreachable from one another, especially in directed networks. The aforementioned centrality metrics are mainly suitable for unweighted networks, but weights on edges contain additional valuable information. Laplacian Centrality was introduced for weighted networks by [28]. In this metric, the so-called Laplacian energy is calculated for a given network, and the importance of a node is determined by the reduction in Laplacian energy when that node is removed from the network. VoteRank, introduced by [29], determines the ranking of nodes in a network using a voting scheme. In VoteRank, each node votes for all its incoming neighbors, and the node with the highest votes is iteratively selected. It was first used to identify influential spreaders in complex networks and tested on real datasets in terms of both the affected scale and spreading rate. PageRank, first introduced by [30] to rank web pages, found applications in various areas, including assessing the importance of nodes for web information retrieval [31] and call graphs [32]. LeaderRank, introduced by [33], determines the ranking of users in social networks. It is similar to PageRank but differs in its high tolerance to noisy data and faster convergence ability.

In addition to the well-known centrality metrics, various novel heuristic methods have been proposed. He et al. [34] introduced a novel selection scheme for critical nodes, eliminating similarity during the influence counts. Fei et al. [35] proposed the inverse-square law to detect the most influential nodes in complex networks. The inverse-square law dictates that interactions diminish linearly as the square of distance increases. Fei et al. [35] extended this concept for complex networks and quantified node influence by aggregating attractions with other nodes. Wang et al. [36] introduced seed exclusion and centripetal centrality methods to detect vital nodes in social networks. Centripetal centrality assesses a node's influence by incorporating its global, local, and semi-local details, producing a more comprehensive result. The proposed seed exclusion method was later devised within the framework of centripetal centrality. Pu et al. [37] introduced a concept called fuzzy local dimension (FLD) to identify influential nodes in complex networks, where nodes with a higher FLD are considered to possess greater influence. Huang et al. [38] introduced a graph partition approach called PartitionRank, which accounts for the characteristics of social media, specifically microblogging scenarios. In microblogging, users can freely choose whom to follow, unlike other social networks that require mutual consent for connections. Shang et al. [39] proposed the distance gravity method, capable of capturing the dynamic interaction between nodes in networks. Later, Curado et al. [40] and Xu and Dong [41] expanded the distance gravity model proposed by Shang et al. [39], introducing the random walk gravity centrality metric and the communicability-based gravity model, respectively. The former improves the detection of key nodes in complex networks by using effective distances in a gravity model and return random walks to highlight community structures and enhance node centrality. The latter contributes to the literature by addressing the heterogeneity of node influence radii and incorporating the impact of node locations on connectivity within networks. Venunath et al. [42] introduced a golden ratio optimization approach to detect the most influential users in social media.

Jiang et al. [43], however, have expanded the methodology for identifying influential nodes beyond heuristic approaches, focusing on the mathematical formulation for identifying the most influential nodes within a given context. Specifically, Jiang et al. [43] aim at optimally identifying influential nodes in directed networks by developing an optimization model. Through a series of experiments, they validate the efficiency of their proposed model when dealing with multiple influential nodes. Their developed model requires considerable memory allocation and computational time. To the best of our knowledge, no study has been proposed for the purpose of improvement in this regard. Hence, this paper initially presents a novel edge-based formulation to solve this problem with reduced computational demands. Later, an equivalent node-based formulation is introduced to address the challenges posed by large networks. A methodology is applied to selectively eliminate redundant constraints in node-based formulation, thereby further reducing its computational complexity. In addition, the aforementioned formulations aim at maximizing the total amount of pair influence. Here, some influential nodes share common neighbors, leading to an inherent issue of overlapping influence during the influence calculation. To address this, an alternative formulation with the objective of maximizing the number of influenced nodes is introduced. Under this formulation, it suffices for a node to have just one influential neighbor to be considered as an influenced node. This approach aims to reduce the overlapping influence, potentially leading to more extensive propagation of influence throughout the network.

Briefly, the contributions of this study can be summarized as follows.

- First, this paper revisits the recent mathematical formulation of [43] and revises it a way that saves substantial computational time. Hence, the most influential nodes even for very large networks can be easily identified within a reasonable computational time frame,
- Second, this paper introduces a novel alternative optimization model built upon the perspective of eliminating overlapping counts in the influence calculation. This formulation accelerates the spread of influence propagation throughout the network without causing additional computational burden.

The subsequent sections of this paper are organized as follows. Section 2 introduces the optimization formulations, along with a constraint reduction methodology to accelerate the identification of influential nodes in directed networks. Section 3 outlines comprehensive computational experiments. Finally, Section 4 summarizes the paper with concluding remarks and proposes suggestions for potential avenues of future research.

2 Problem formulations

Let G be a directed network comprising node set denoted as N and arc set denoted as E. To depict network topology and capture interconnectivity information, an adjacency matrix is employed, comprising an $|N| * |N|$ matrix, with "$|N|$" denoting the number of nodes within the network. Let $|E|$ be the number of edges in the network. Each entry a_{ij} in the adjacency matrix A signifies the adjacency relationship between nodes i and j; it assumes a value of 1 to denote a connection between nodes i and j, and 0 otherwise. The variable x_{ij} is a binary variable that represents the presence or absence of an incoming edge from the influential node i to any node j. Likewise, the variable y_i serves as a binary variable, denoting the selection or non-selection of node i as an influential node. Here, the selection of influential nodes is constrained by a finite availability of resources, represented by an upper limit denoted as K. The aforementioned decision variables are formally defined in the following manner.

$$x_{ij} = \begin{cases} 1, \text{if node j is influenced by node i} \\ 0, \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1, \text{if node i is selected as an influential node} \\ 0, \text{otherwise} \end{cases}$$

With the above definitions and notations in mind, the influence maximization model developed by [43], hereafter referred to as **IM^{A}**, is presented as follows:

$$\text{Max} \sum_{i \in N, j \in N} a_{ij}\, x_{ij} \tag{1}$$

$$\sum_{i=1}^{n} x_{ij} \leq K, \quad \forall\, j \in N \tag{2}$$

$$x_{ij} \leq y_i, \quad \forall\, i \in N, j \in N \tag{3}$$

$$x_{ij} \leq a_{ij}, \quad \forall\, i \in N, j \in N \tag{4}$$

$$x_{ij} + y_i + y_j \leq 2, \quad \forall\, i \in N, j \in N \tag{5}$$

$$\sum_{i \in N} y_i \leq K \tag{6}$$

$$x_{ij}, y_i \in \{0,1\}, \quad \forall\, i \in N, j \in N \tag{7}$$

Objective function (1) in model IM^{A} maximizes the influence exerted by a designated set of influential nodes within a network. Constraints (2) ensure that each node is influenced by at most K influential nodes. Constraints (3) focus on spreading influence through the influential nodes rather than other nodes. Constraints (4) guarantee that the influence is limited between neighborhoods. Constraints (5) prohibit the involvement of interactions between sets of influential nodes in the influence calculation. Constraint (6) establishes an upper limit of K on the number of influential nodes. Finally, Constraints (7) force that the decision variables exhibit a binary nature, meaning they must possess 0 or 1 value at optimality. Model IM^{A} includes: (1) $|N|$ of Constraints (2), (2) $|N| * |N|$ of Constraints (3), (3) $|N| * |N|$ of Constraints (4), (4) $|N| * |N|$ of Constraints (5), (5) one Constraint (6), and (6) $|N| * |N| + |N|$ of binary variables.

Alternatively, we introduce a novel edge-based formulation, hereafter referred to as **IM^{E}**, to speed up the resolution of the identical problem. IM^{E} employs the identical notations to IM^{A} and exhibits the following structural framework:

$$\text{Max} \sum_{(i,j) \in E} x_{ij} \tag{8}$$

$$x_{ij} \leq y_i, \quad \forall\, (i,j) \in E \tag{9}$$

$$x_{ij} \leq 1 - y_j, \quad \forall\, (i,j) \in E \tag{10}$$

$$\sum_{i \in N} y_i \leq K \tag{11}$$

$$x_{ij} \geq 0, \forall\, (i,j) \in E, y_i \in \{0,1\}, \forall\, i \in N \tag{12}$$

Objective (8) of IM^{E} is identical to Objective (1) of IM^{A}, but differs in that it is formulated based on the edge set E, allowing the removal of the parameter a_{ij} in the objective. Constraints (9)-(10) exactly serve the same purpose akin to that of Constraints (2)-(5). Note that Constraints (9)-(10) are edge-based constraints, i.e., the number of constraints is proportional to the number of edges in the network. Constraint (6) and (11) are exactly the same. Even though the decision variable x is now relaxed in Constraint (12) of IM^{E} in order to further shorten the model run time, it holds the binary nature at optimality, as proved in Lemma 2 given below. IM^{E} includes: (1) $|E|$ of Constraints (9), (2) $|E|$ of Constraints (10), (3) one Constraint (11), and (4) $|E|$ of positive variables and $|N|$ of binary variables.

Lemma 1: IM^{E} is a NP-complete problem.

Proof. IM^{E} has the following structure. Universe $\mathcal{L}$ with $|\mathcal{L}| = N$. Define a collection of sets $\mathrm{N} = \{E_1, E_2, \ldots E_{|E|}\}$, where each $E_i \subseteq \mathcal{L}$ for all i and integer $K \leq N$. A feasible set includes $\Psi \in [|E|]$ (where, $[|E|]$ refers to 1,2, ...,$|E|$) such that $|\Psi| \leq K$. The aim is to maximize the cardinality of the union sets of E_i, $\max |\cup_{i \in \Psi} E_i|$. Notice that this problem is a version of the "set covering problem", known as NP-complete problem. Hence, this problem falls into the category of NP-complete problems, thus categorizing IM^{E} as an NP-complete problem. This completes the proof.

Lemma 2. In model IM^{E}, an optimal solution is present in which the variables x_{ij} must hold binary nature.

Proof. Observe that the variables y_i and x_{ij} are defined as binary variables (0 or 1) and positive variables, respectively in Constraints (12). Keeping this fact in mind that Constraints (9) enforce that if y_i is 0 (implying that node i is not chosen as an influential node), then x_{ij} must also be 0 for all edges $(i,j) \in E$. If y_i is 1 (implying that node i is chosen as an influential node), it allows x_{ij} to take any positive value within the interval [0,1]. Similar to the previous constraint, Constraints (10) imply that if y_j is 1, then x_{ij} must be 0 ($x_{ij} \leq 1 - 1 = 0$). If y_j is 0, it allows x_{ij} to take any positive value within the interval [0,1]. Based on these facts, any positive variable x_{ij} must hold the following: either case (1) $x_{ij} = 0$ or case (2) $0 < x_{ij} \leq 1$. In case (1), x_{ij} obviously maintains its binary nature. In case (2), since model IM^{E} is a maximization problem with an objective equal to the sum of the x_{ij} variables, any positive x_{ij} variable tends to always take its maximum value of 1 within Constraints (9)-(10) instead of any fractional value in the range 0 to 1. Given these considerations, at optimality, the x_{ij} variables are inherently driven to take binary values. This completes the proof.

Lemma 3. IM^{E} always yields the optimal objective value.

Proof. From Lemma 1, we have that any strictly positive variable x_{ij} takes the value of 1 at optimality in IM^{E} because of its objective function structure. As the objective functions of both IM^{A} and IM^{E} are the equivalent, showing the constraints of IM^{E} always produce the true solution space for x_{ij} is enough to prove this lemma. Consider a network whose nodes i and j are connected by a direct edge from node i to node j. Here, only four distinctive cases are delineated:

Case (1) : Node i is designated as the influenced node (i.e., $y_i = 1$), while node j remains uninfluential node (i.e., $y_j = 0$). For this scenario, since node i exerts influence on node j, by the definition of x_{ij} variable, the variable x_{ij} must take the value of 1. From Constraints (9) and (10), we obtain $x_{ij} \leq 1$ and $x_{ij} \leq 1 - 0 = 1$, respectively. Thus, within the framework of these two constraints, x_{ij} must take the value of 1, which aligns with its intended value,

Case (2) : Conversely, node j is designated as the influenced node, while node i remains uninfluential node. For this scenario, as node j cannot be influenced by an uninfluential node i, the variable x_{ij} must be 0. From Constraints (9) and (10), we obtain $x_{ij} \leq 0$ and $x_{ij} \leq 1 - 1 = 0$, respectively. Thus, within the framework of these two constraints, x_{ij} should adopt the value of 0, which again aligns with its intended value,

Case (3) : Both node i and node j are concurrently designated as influenced nodes. For this scenario, by definition in Constraint (5), the variable x_{ij} must be set to 0. From Constraints (9) and (10), we obtain $x_{ij} \leq 1$ and $x_{ij} \leq 1 - 1 = 0$, respectively. So, the solution space satisfying both Constraints (9) and (10) forces x_{ij} to take the true value of 0,

Case (4) : Alternatively, neither node i nor node j is designated as influenced nodes. By definition, the variable x_{ij} must be set to zero, as observed in Constraint (3). From Constraints (9) and (10), we obtain $x_{ij} \leq 0$ and $x_{ij} \leq 1 - 0 = 1$, respectively. So, the solution space satisfying both Constraints (9) and (10) obviously results in 0, the true value of x_{ij}. This completes the proof.

Lemma 3 shows that Constraints (9)-(10) of IM^{E} are implicitly identical to Constraints (2)-(5) of IM^{A}. Hence, IM^{E} presents a noteworthy reduction in the number of constraints, as well as variables relative to its predecessor, IM^{A}. Specifically, whereas model IM^{A} has $3|N|^2 + |N| + 1$ constraints and $|N|(|N| + 1)$ binary variables, IM^{E} has $2|E| + 1$ constraints and $|E| + |N|$ decision variables, of which only $|N|$ number of variables are defined as binary variables.

The aforementioned modeling enhancement is expected to result in a substantial decrease in solution time for IM^{E} when compared to IM^{A}. However, an alternative node-based formulation is also introduced for the sake of its scalability for larger networks, hereafter referred to as **IM^{N}**. This compact formulation IM^{N} uses a newly defined positive variable z_i, which represents the total number of influential neighbor nodes of node i. Alternatively, z_i can be defined as the total amount of influence exerted on node i by its neighbors. Let ϑ_i^{out} and ϑ_i^{in} be out-degree and in-degree of node i, respectively. Let $\mathcal{L}$ be a sufficiently large positive number, which can be set as the maximum out-degree in the network (i.e., $\mathcal{L} = \max_{i \in N}\{\vartheta_i^{out}\}$).

IM^{N} reads as follows:

$$\text{Max} \sum_{i \in N} z_i \tag{13}$$

$$z_i + \mathcal{L} y_i \leq \mathcal{L}, \quad \forall i \in N \tag{14}$$

$$z_i \leq \sum_{j:(j,i) \in E} y_j, \quad \forall i \in N \tag{15}$$

$$\sum_{i \in N} y_i \leq K \tag{16}$$

$$z_i \geq 0, y_i \in \{0,1\}, \forall i \in N \tag{17}$$

The objective function (13) aims at maximizing the total amount of influence, as in Objective (1) and (8). Constraints (14) guarantee that if node i is selected as an influential node, then z_i must take the value of 0. This implies that the already influenced node i cannot be influenced by its neighbors. Conversely, if node i is not selected as an influential node, then Constraint (14) becomes a non-binding constraint for given i. For the cases where Constraints (14) are non-binding (i.e., $y_i = 0$), Constraints (15) become binding constraints and ensure the value of z_i is restricted by its total number of influential neighbor nodes. Constraint (16) is the same as Constraint (11). Constraints (17) provide the nature of variables. The superiority of IM^{N} over IM^{E} lies in its node-based constraints. To elaborate further, the number of constraints in the IM^{N} model exhibits a linear relationship with the size of the nodes of the network, resulting in a more scalability advantage. Specifically, IM^{N} includes only $2|N|+1$ constraints, $|N|$ binary and $|N|$ positive variables.

Note that Objective (13) and Constraints (15) can be run over $i \in N$ such that $\vartheta_i^{in} > 0$. This means that node i has not incoming edge that allows it to be influenced by its immediate neighbors. Constraints (14) can be run over $i \in N$ such that $\vartheta_i^{in} > 0$ and $\vartheta_i^{out} > 0$, which implies that node i should have at least one incoming and outgoing edge; otherwise, Constraints (14) become unbinding constraints. Such effort further eliminates the redundant terms in objective function and the redundant set of constraints in Constraints (14)-(15). Similarly, the summation term of Constraint (16) can be defined over $i \in N$ such that $\vartheta_i^{out} > 0$ because there is no chance for any node j to be influenced by node i if node i has no outgoing edges. For the sake of space, we do not rewrite the formulation after these reductions from scratch, but instead define this reduced-version of IM^{N} as **IM^{Nr}** in our computational testing.

Within the influential nodes found by the aforementioned formulations, it is possible that some influential nodes share common neighbors, leading to an inherent issue of overlapping influence. Namely, the substantial similarity between the neighbors of these nodes may hinder their effectiveness in

influencing the network on a large scale. To address this and potentially enhance the initial selection of influential nodes, we introduce a third formulation, named hereafter as **IM^{O}**. Here, the superscript "O" is used to refer to overlapping. The proposed formulation utilizes a new variable, w_i, defined as follows:

$$w_i = \begin{cases} 1, \text{if node i is influenced by at least} \\ \text{one of its influential neighbors} \\ 0, \text{otherwise} \end{cases}$$

Model IM^{O} is outlined as follows.

$$\text{Max} \sum_{i \in N} w_i \tag{18}$$

$$w_i + y_i \leq 1, \quad \forall i \in N \tag{19}$$

$$w_i \leq \sum_{j:(j,i) \in E} y_j, \quad \forall i \in N \tag{20}$$

$$\sum_{i \in N} y_i \leq K \tag{21}$$

$$w_i \geq 0, y_i \in \{0,1\}, \forall i \in N \tag{22}$$

Note that, to best of our knowledge, this solution modelling is being utilized for the first time in INIP. The objective function (18) now aims at maximizing the number of influenced nodes, which eliminates the issue of overlapping. Constraints (19) guarantee that when a node is selected as an influential node, it is precluded from simultaneously being considered as an influenced node. Constraints (20) guarantee that a node can be an influenced node, providing that at least one of its neighbors is an influential node. Whereas Constraints (22) share similarities with Constraints (17) but entail minor variations in their specific criteria, Constraint (21) is identical to Constraint (16). To enhance computational efficiency of model IM^{O} in our computational testing, we redefine the search space for Objective (18) and Constraints (19)-(20) to include nodes $i \in N$ with $\vartheta_i^{in} > 0$, while Constraint (21) is adjusted for nodes $i \in N$ where $\vartheta_i^{out} > 0$.

Lemma 4. In model IM^{O}, an optimal solution is present in which the variables w_i must hold binary nature even though they are relaxed for the sake of computational run time.

Proof. For each node i, w_i is constrained to have a maximum value of 1, as stipulated by Constraints (19) in IM^{O}. Given that, w_i satisfies the following conditions:

1. If y_i equals 1 (indicating that node i is chosen as an influential node) or $\sum_{j:(j,i) \in E} y_j$ equals to 0 (indicating that none of neighbors of node i are influential nodes), or if both conditions are met simultaneously, then w_i must be set to 0 as dictated by Constraints (19)-(20). In these cases, w_i maintains its binary natüre,
2. However, if y_i equals 1 for node i and at least one of neighbors of node i is an influential node, the model assigns w_i a value of 1 as the objective of IM^{O} is to maximize the summation of w_i. Hence, w_i again retains its binary structure, even though it is relaxed in the model. This completes the proof.

Note that although these reductions do not overcome the inherent issue of the scalability of the Mixed Integer Programming models IM^{E} and IM^{O} (i.e., their NP-completeness proved by Lemma 1), the computational testing shows the noticeable performance improvement achieved by the proposed models. In other words, computational testing demonstrates that the reduced complexity in terms of constraints and variables translates to better scalability in memory usage and solution speed. Hence, the proposed models can handle larger instances of the problem more effectively than the model IM^{A} from the literature, which has more constraints and variables.

3 Computational tests

The computational tests were conducted on a personal computer equipped with an Intel i7-11800H processor running at 2.3 GHz and 16 GB of RAM. To ensure a fair and consistent comparison, all experiments were conducted on the same platform. The GAMS platform was employed to write and implement the mathematical formulations, and CPLEX 12.6.2.0 was selected as the solver to solve the models, with its default settings, except for the optimality gap, which was set to zero. This controlled setup eliminates variability due to differing hardware or software environments, ensuring that the observed differences in computation times are solely attributable to the models themselves. A maximum CPU time limit of 18,000 seconds (s) was enforced during the computational testing. If an instance was not successfully solved within this limit, then it was indicated in the tables with a ">" symbol. The results obtained from the studied methods were evaluated in simulation tests conducted using Python 3.10. The computational testing consists of two subsections as given below.

3.1 The performance of optimization models

The first section scrutinizes the performance of the optimization models, specifically focusing on their solution speed, aiming to show how the proposed models (IM^{E}, IM^{N} and IM^{Nr}) shorten solution times compared to their counterpart IM^{A} of [43]. The computational performance of IM^{O} is also presented in this section.

Table 1 summarizes the properties of networks utilized for evaluating the performance of all models. In Table 1, $|N|$, $|E|$, $\langle k \rangle$ and $\mathcal{L}$ respectively represent the node count, edge count, average degree, and maximum out-degree in the network, with the exclusion of any single-edge cycles. In addition, α, β, and γ represent the counts of nodes having outgoing edges, incoming edges, and both outgoing and incoming edges, respectively. Networks Cage6, Bcspwr06, Hi2010, Shyy161, Ford2 and SNAP/email-EuAll were obtained from [44]; networks Anheim, GoldCoast and ChicagoRegional were obtained from [45]. Note that if parameters α, β and γ are found to be equal to |N| for a given network, the performance of IM^{Nr} is not separately reported for that network since it exhibits identical performance to IM^{N}. In Tables 2 and Table 3, the set of influential node values K are varied from 1 to 10.

Table 2 compares the performance of the models by using networks having less than 500 nodes. Observe in Table 2 that IM^{E} and IM^{N} consistently outperform IM^{A}, but the performance improvement, in terms of reduced solution time, is relatively modest, typically less than one second on average. This, of course, is due to the fact that working on small networks masks the performance improvement between models. Table 3 presents a comparison of model performances using networks with between 1,000 nodes 10,000 nodes.

Table 1. Network properties

Network	$\|N\|$	$\|E\|$	$\langle k \rangle$	$\mathcal{L}$	α	β	γ
Cage6	93	692	14.88	12	93	93	93
Anheim	416	914	4.39	6	416	416	416
Bcspwr06	1454	1923	2.65	11	1083	1003	632
GoldCoast	4783	11140	4.65	6	4783	4783	4783
ChicagoRegional	10959	20019	3.65	6	7580	8963	5584
Hi2010	25016	62063	4.96	57	20183	20055	15222
Rajat26	50932	200734	7.88	3400	50932	50932	50932
Shyy161	76480	278882	7.29	5	76480	76480	76480
Ford2	100196	222246	4.44	29	93920	92374	86098
SNAP/email-EuAll	265214	418956	3.16	7631	225137	74445	34573

Table 2. Comparisons of the models using networks Cage6 and Anheim

	Cage6					Anheim			
		Time (s)					Time (s)		
K	Obj	IM^A	IM^E	IM^N	K	Obj	IM^A	IM^E	IM^N
1	12	0.13	0.10	0.09	1	6	1.53	0.22	0.20
2	22	0.14	0.10	0.09	2	12	1.53	0.10	0.09
3	33	0.23	0.10	0.09	3	18	1.51	0.10	0.09
4	44	0.23	0.10	0.09	4	23	1.53	0.10	0.10
5	54	0.24	0.10	0.10	5	28	1.52	0.10	0.10
6	63	0.15	0.13	0.10	6	33	1.62	0.10	0.09
7	72	0.19	0.15	0.11	7	38	1.52	0.10	0.09
8	80	0.44	0.19	0.21	8	43	1.55	0.10	0.09
9	89	0.37	0.37	0.23	9	48	1.57	0.10	0.09
10	98	0.38	0.33	0.21	10	53	1.51	0.10	0.09

Table 3. Comparisons of the models using networks Bcspwr06 and GoldCast

	Bcspwr06					GoldCast			
		Time (s)					Time (s)		
K	Obj	IM^A	IM^E	IM^N	K	Obj	IM^A	IM^E	IM^N
1	9	102.81	0.30	0.32	1	6	>	2.31	1.59
2	17	91.98	0.21	0.20	2	12	>	2.20	0.93
3	24	92.86	0.30	0.20	3	18	>	2.54	0.96
4	31	84.87	0.21	0.20	4	23	>	2.16	0.87
5	37	92.91	0.30	0.20	5	28	>	2.25	0.88
6	43	92.38	0.22	0.20	6	33	>	2.16	0.95
7	49	87.07	0.31	0.20	7	38	>	2.33	0.95
8	55	93.09	0.31	0.20	8	43	>	2.28	0.95
9	61	93.04	0.31	0.20	9	48	>	2.19	0.94
10	67	94.55	0.20	0.19	10	53	>	2.37	0.95

As seen in Table 3, when the network size grows, the performance improvement between models become more apparent compared to Table 2. For example, in Bcswpr06 network, the average required computational times to obtain solutions are 91.42, 0.27 and 0.21 seconds for IM^A, IM^E and IM^N, respectively. As noticed, the performance of IM^A is notably inferior, while the performance of IM^E and IM^N is nearly indifferentiable, showing very similar results. For GoldCast network, IM^A could not return the optimal solution within the 5-hour (18,000 seconds) time frame, whereas IM^E and IM^N respectively provided the optimal solution in 2.28 and 1.00 seconds on average.

Tables 4, 5 and 6 compare the performance of the models using networks having more than 10,000 nodes. The networks in these tables are arranged in ascending order based on their sizes. These tables only evaluate the performance of the models based on K=5, 10, 50, 100, 250 and 500 values for the sake of space. None of the networks in these tables were solved by IM^A within a predefined time frame of 5-hour. Observe in Tables 4, 5 and 6 that IM^{Nr} always performed the best when applied. IM^N always performed better than IM^E for all networks. For network ChicagoRegional in Table 4, IM^{Nr} returned solutions more than three times faster than IM^E on average. Likewise, when considering the Hi2010 network, IM^{Nr} consistently delivers solutions in a significantly shorter average time, averaging 17.49 seconds, compared to IM^E, which has an average solution time of 53.01 seconds. Also, IM^N shows superior performance compared to IM^E for the networks in Table 3. With increasing network size, the distinction between IM^N and IM^E becomes increasingly pronounced. For example, for the largest network SNAP/email-EuAll, IM^N and IM^{Nr} obtain the solution in 3212.21 and 750.78 seconds on average respectively, while IM^E obtains the solution in 8321.86 seconds. Put differently, IM^A solves on average 2.6 and 11 times slower than IM^N and IM^{Nr}, respectively. This result can also be interpreted as IM^{Nr} obtaining the optimal solution, on average, 7,571.08 seconds earlier than IM^E.

Table 4. Comparisons of models using networks ChicagoRegional and Hi2010

	ChicagoRegional						Hi2010				
		Time (s)						Time (s)			
K	Obj	IM^A	IM^E	IM^N	IM^{Nr}	K	Obj	IM^A	IM^E	IM^N	IM^{Nr}
5	30	>	10.33	3.67	3.20	5	255	>	54.73	21.64	16.94
10	60	>	10.12	3.64	3.21	10	446	>	50.98	21.33	16.85
50	263	>	10.00	3.84	3.24	50	1412	>	52.01	22.13	17.93
100	479	>	10.11	3.77	3.24	100	2253	>	52.18	21.15	17.39
250	1079	>	10.17	3.85	3.23	250	4139	>	51.65	21.87	17.43
500	2079	>	10.32	3.86	3.34	500	6534	>	56.54	22.03	18.42

Table 5. Comparisons of the models using networks Rajat26 and Shyy161

	Rajat26					Shyy161			
		Time (s)					Time (s)		
K	Obj	IM^A	IM^E	IM^N	K	Obj	IM^A	IM^E	IM^N
5	8741	>	235.56	80.29	5	25	>	594.38	176.79
10	11779	>	262.20	80.25	10	50	>	531.93	188.63
50	21472	>	227.30	78.05	50	250	>	537.83	205.60
100	23710	>	332.63	81.23	100	500	>	577.92	216.26
250	28223	>	242.02	83.78	250	1250	>	548.94	223.12
500	32796	>	245.04	89.45	500	2500	>	962.60	197.07

Table 6. Comparisons of models using networks Ford2 and SNAP/email-EuAll

	Ford2						SNAP/email-EuAll				
		Time (s)						Time (s)			
K	Obj	IM^A	IM^E	IM^N	IM^{Nr}	K	Obj	IM^A	IM^E	IM^N	IM^{Nr}
5	134	>	815.63	304.41	269.28	5	4218	>	10207.49	3005.24	641.80
10	244	>	830.00	293.84	272.24	10	7477	>	10115.44	3010.17	644.06
50	878	>	809.27	295.73	268.07	50	23442	>	7368.62	3001.24	696.80
100	1482	>	837.75	296.16	269.96	100	36213	>	7847.02	3024.85	731.39
250	2955	>	832.45	298.72	280.56	250	58355	>	7339.13	3636.20	887.24
500	5205	>	837.49	299.48	276.05	500	74037	>	7053.45	3595.55	903.39

In brief, Tables from 1 to 6 show that the newly developed models are capable of achieving provably optimal solutions across a range of test networks in significantly reduced time when compared to IM^A of [43]. In addition, the node-based formulations (i.e., IM^N and IM^{Nr}) almost always outperform the edge-based formulation IM^E, as expected. Finally, the constraint reduction strategy, as applied to IM^N (referred to as model IM^{Nr}), always reduces the needed solution time across various networks, including ChicagoRegional, Hi2010, Ford2 and SNAP/email-EuAll. The main reason for the differences in CPU time is that both IM^E and IM^N greatly reduce the search space compared to IM^A, which requires generating many redundant sets of constraints. This fact holds true for all computational testing conducted throughout the paper. However, the developed models are still limited as they are MIP models, which inherently face the scalability issue (i.e., NP-Completeness), especially for large real-word networks including millions of nodes and arcs.

Finally, we explore how using model IM^O instead of the other models impacts the time it consumes to find optimal solutions. Hence, Table 7 details the average solution times for all optimization models, representing the mean solution times corresponding to K values from the previous tables. Observe in Table 7 that IM^O delivers solutions quicker than the other models in general. As noted, IM^{Nr} and IM^O are similar in many respects, with slight differences in Constraints (14) and (19) and the constraint reduction strategy applied to these constraints, respectively. These differences cause relatively minor changes in computational time compared to comparisons with the other models, as shown in Table 7.

3.2 Comparison of the methods in minimizing spread

This section conducts a comparative analysis of the methods (i.e., Degree-centrality (DC), IM^A, IM^O) with regard to the influence time metric, which measures the speed at which the initially selected influential nodes influence the entire network. Since IM^A, IM^E, IM^N, and IM^{Nr} all yield identical sets of influential nodes, we exclusively included IM^A and IM^O in our testing to avoid redundancy. The tests were performed on two small size networks IMB32 and GD95a, obtained from [44]. Network IBM32 contains 32 nodes and 94 edges; network GD95a contains 36 nodes and 57 edges. Note that the objectives of the tested methods are not focused on spread-related goals, such as slowing down the spread, which can be measured using the time metric to influence all nodes. Hence, to assess the efficacy of these methods, we estimate the time required to influence all nodes by means of simulations following the initial selection of influential nodes from networks specified by the examined methods.

In the infection-related literature, various studies generally have employed one of the two models: SI (Susceptible-Infectious) and SIR (Susceptible-Infectious-Recovered). In the context of INIP, SI model means that the influenced nodes remain influenced indefinitely without recovery. Conversely, in the SIR model, influenced nodes exhibit a fixed probability of recovering from the influence.

Table 7. Comparisons of all models in terms of solution return time.

Network	IM^A	IM^E	IM^N	IM^{Nr}	IM^O
Cage6	0.25	0.17	0.13	0.13	**0.09**
Anheim	1.54	0.11	0.10	**0.10**	**0.10**
Bcspwr06	92.56	0.27	0.21	0.21	**0.18**
GoldCoast	>	2.28	1.00	1.00	**0.90**
ChicagoRegional	>	10.18	3.77	3.24	**3.18**
Hi2010	>	53.02	21.69	17.49	**17.29**
Rajat26	>	257.46	82.18	82.18	**81.42**
Shyy161	>	625.60	201.25	201.25	**161.20**
Ford2	>	827.10	298.06	**272.69**	**272.69**
SNAP/email-EuAll	>	8321.86	3212.21	750.78	**669.64**

*: The numbers highlighted in bold indicate the lowest average solution time for a given network.

Although SIR provides a practical and realistic estimate suitable for a wide range of spreading scenarios, SI is used in the simulation testing. This decision stemmed from the challenge of estimating time to infect all non-influential nodes in SIR as nodes recovering from the influence potentially cause the spread to cease before influencing all non-influential nodes.

In SI simulation testing, random numbers are produced for every edge at each stage, and influential nodes transmit the influence to uninfluenced neighbors if the random number falls below a predefined transmission threshold (symbolized hereafter as p). This process continues until all nodes become influential nodes, at which point we record the expected time it took to influence all nodes. The expected average time is symbolized hereafter as E(T). In testing, SI simulation test is replicated 250,000 times to ensure unbiased E(T) results.

In contrast to the assertion in paper [43], the monotonicity of optimization models (e.g., IM^A) is not consistently equal to 1. Put differently, diverse optimal results can be attained depending on the factors such as the utilized platform, solver, and so on. For example, 2, 4 and 1 different alternative optimal solutions exist for DC, IM^A and IM^O, respectively for the K=3 scenario. In this regard, to ensure a fair comparison, every potential optimal solution scenario is generated for all three methods across each K value. The resulting average times of E(T), a metric to measure the expected speed of the influence spread, are then recorded based on the initial K-node selection from the methods. Noticed that the lower the E(T) value, the more effective the method is at influencing all nodes in the network.

Observe in Figure 1 that model IM^O always fully influence IBM32 network faster than both IM^A and DC for every value of K. While DC generally exhibits the worst performance, there is only one instance (i.e., K=1) where DC, IM^A and IM^O exhibit equal performance in terms of average E(T) values. Although not presented in Figure 1, the slowest and fastest E(T) times for all three methods were examined for each K value in IBM32 network as well. We observe that while DC generally exhibits the worst performance, there is only one instance (i.e., K=2) where IM^A outperforms IM^O in terms of maximum E(T) value.

Observe in Figure 2 that IM^O always demonstrates the top performance in terms of average E(T) times, whereas DC consistently exhibits the poorest performance for network GD95a. In network GD95a, there is no instance that IM^A outperforms IM^O with respect to average E(T) times. For this network, IM^O reduced E(T) by approximately to 24.68 % and 5.27 % on average compared to DC and IM^A, respectively. Even though not presented in Figure 2, the slowest and fastest E(T) times for all three methods were recorded for each K value in GD95a network as well. We observe that IM^A surpasses IM^O in a few cases (i.e., K=2, 3, and 4) regarding maximum E(T) values but consistently falls short in terms of minimum and average E(T) values.

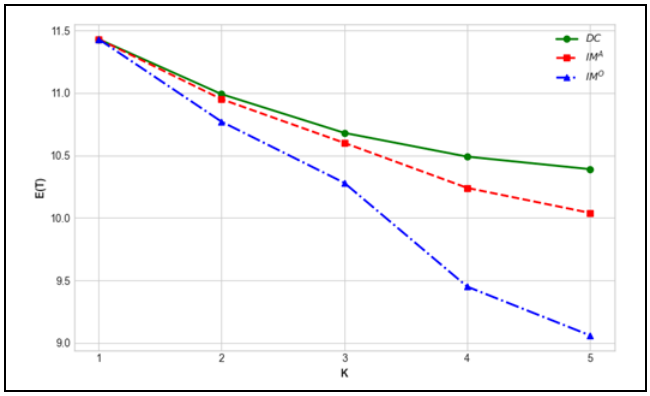

Figure 1. Comparisons of the methods in terms of the average E(T) for IBM32 network.

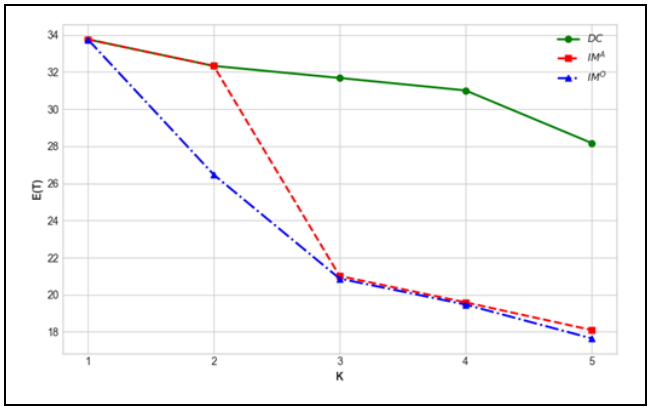

Figure 2. Comparisons of the methods in terms of the average E(T) for GD95a network

Note that the parameter p was set to 0.25 in the simulation testing. In the following Figures 3, 4 and 5, p values are varied between 0.25 and 0.90 to observe the response of DC, IM^A and IM^O to varying p values. The figures demonstrate that as the p value increases, the time required to influence the nodes decreases. This outcome is expected, as a higher p value corresponds to an increased likelihood of influencing neighboring nodes for all three methods.

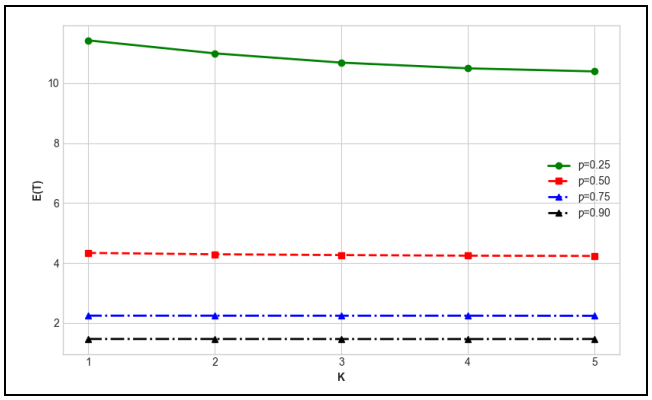

Figure 3. E(T) of DC based on various p values for IBM32.

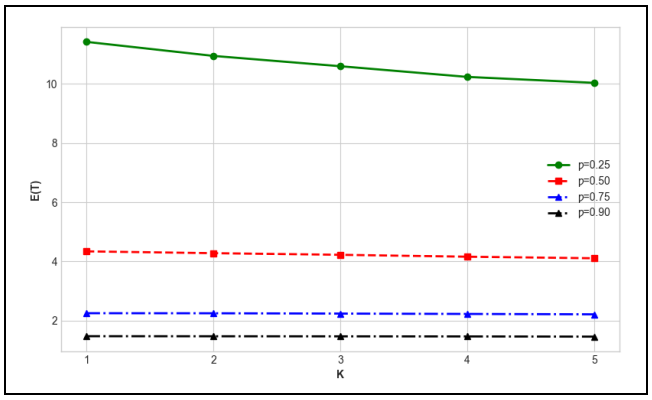

Figure 4. E(T) of IM^E based on various p values for IBM32.

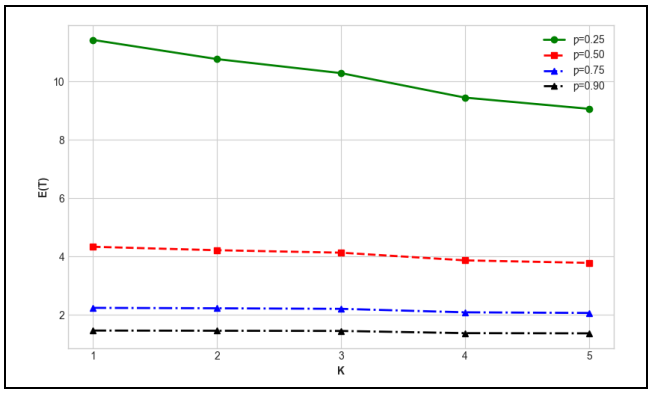

Figure 5. E(T) of IM^O based on various p values for IBM32.

As seen in the following figures, when the K value is increased from 1 to 5, the differences in E(T) values become more pronounced, particularly at lower p values. This outcome is intuitive: as p increases, the likelihood of transferring influence to other nodes rises. Consequently, the effect of the initial influential node selection's position and size on E(T) weakens with higher p values. Since IM^A and IM^O are scenario-based approaches that simultaneously evaluate the impact of multiple influential nodes, the enhancement in performance becomes more apparent as K increases from 1 to 5, especially for IM^O, the best performing model when simultaneously determining the location of K influential nodes.

4 Conclusion

This paper explores the critical research area of identifying influential nodes in complex networks based on a scenario-based approach where multiple nodes are simultaneously selected as influential nodes. Hence, novel formulations are introduced to efficiently identify influential nodes in directed networks in a short period of time. First two novel formulations (i.e., IM^E and IM^N) aim to reduce solution time compared to their counterpart in the literature, whereas an alternative formulation (i.e., IM^O) eliminates overlapping influence effects of the previous formulations. Based on the results from Table 2 to Table 6, the proposed models, IM^E and IM^N, improve upon IM^A of [43], enabling quicker identification of influential nodes with reduced computational resources. Similarly, based on the results from Figures 1 and 2, along with Table 7, it is fair to conclude that IM^O outperforms the other models in terms of both faster solution discovery and the performance of the obtained solution's influence propagation based on SI simulation. These findings demonstrated the power of the proposed formulations for determining the most influential nodes. However, the developed models are still limited by their nature as MIP models, which inherently struggle with scalability issues, particularly for large real-world networks containing millions of nodes and arcs. Hence, future research will focus on proposing innovative heuristic approaches to address the challenges in INIP with various objectives. Specifically, we will propose a novel path-based algorithm and compare it with state-of-the-art methods in the literature. The comparison will cover various metrics, including average new infections and time to influence half of the uninfluenced nodes. Additionally, although the optimization models presented here are built upon the context of influence maximization, they can be adapted and applied to various real-world domains such as epidemiology and marketing. For example, in epidemiology, these models can be used to control and mitigate the spread of viruses by identifying key individuals for intervention. By targeting these key nodes within a network, health authorities can optimize vaccination campaigns, quarantine strategies, and other preventive measures to effectively respond to outbreaks. As exemplified, the proposed models have managerial implications for resource allocation and intervention strategies. Hence, in the future, it will also be beneficial to evaluate the proposed methodologies in the aforementioned areas to assess their practical utility.

5 Author contribution statements

In the scope of this study, Gökhan Karaköse in the formation of the idea, the literature review, the mathematical models, the computational experiments, the assessment of obtained results, the spelling and checking the article in terms of content was contributed.

6 Ethics committee approval and conflict of interest statement

"There is no need to obtain permission from the ethics committee for the article prepared".

"There is no conflict of interest with any person / institution in the article prepared".

7 References

[1] Wang Z, Andrews MA, Wu ZX, Wang L, Bauch CT. "Coupled disease-behavior dynamics on complex networks: A review". *Physics of Life Reviews*, 15, 1-29, 2015

[2] Barabási AL, Gulbahce N, Loscalzo J. "Network medicine: a network-based approach to human disease". *Nature Reviews Genetics*, 12(1), 56-68, 2011.

[3] Xiao F, Aritsugi M, Wang Q, Zhang R. "Efficient processing of multiple nested event pattern queries over multi-dimensional event streams based on a triaxial hierarchical model". *Artificial Intelligence in Medicine*, 72, 56-71, 2016.
[4] Xiao F, Zhan C, Lai H, Tao L, Qu Z. "New parallel processing strategies in complex event processing systems with data streams". *International Journal of Distributed Sensor Networks*, 13(8), 1-1, 2017.
[5] Vega-Oliveros DA, da Fontoura Costa L, Rodrigues FA. "Influence maximization by rumor spreading on correlated networks through community identification". *Communications in Nonlinear Science and Numerical Simulation*, 83, 1-13, 2020.
[6] Yan Z, Zhou X, Ren J, Zhang Q, Du R. "Identifying underlying influential factors in information diffusion process on social media platform: A hybrid approach of data mining and time series regression". *Information Processing & Management*, 60(5), 1-20, 2023.
[7] Zhang X, Zhu J, Wang Q, Zhao H. "Identifying influential nodes in complex networks with community structure". *Knowledge-Based Systems*, 42, 74-84, 2013.
[8] Wang Z, Xia CY, Meloni S, Zhou CS, Moreno Y. "Impact of Social Punishment on Cooperative Behavior in Complex Networks". *Scientific Reports*, 3(1), 1-7, 2013.
[9] Chang S, Pierson E, Koh PW, Gerardin J, Redbird B, Grusky D, et al. "Mobility network models of COVID-19 explain inequities and inform reopening". *Nature*, 589(7840), 82-87, 2021.
[10] Yang Y, Wang X, Chen Y, Hu M, Ruan C. "A novel centrality of ınfluential nodes ıdentification in complex networks". *IEEE Access*, 8, 58742-58751, 2020.
[11] Zhang J, Yang C, Jin Z, Li J. "Dynamics analysis of SIR epidemic model with correlation coefficients and clustering coefficient in networks." *Journal of Theoretical Biology*, 449, 1-13, 2018.
[12] Banerjee S, Jenamani M, Pratihar DK. "A survey on influence maximization in a social network". *Knowledge and Information Systems*, 62, 3417-3455, 2020.
[13] Dedeturk BA, Gungor BB. "Evaluation of sub-network search programs in epilepsy-related GWAS dataset". *Pamukkale University Journal of Engineering Sciences,* 28(2), 292-298, 2020.
[14] Zhao Y, Kou G, Peng Y, Chen Y. "Understanding influence power of opinion leaders in e-commerce networks: An opinion dynamics theory perspective". *Information Sciences*, 426, 131-147, 2018.
[15] Cheng CH, Kuo YH, Zhou Z. "Outbreak minimization v.s. influence maximization: an optimization framework". *BMC Medical Informatics and Decision Making*, 20(1), 1-13, 2020.
[16] Chaharborj SS, Nabi KN, Feng KL, Chaharborj SS, Phang PS. "Controlling COVID-19 transmission with isolation of influential nodes". *Chaos, Solitons & Fractals*, 159, 1-11, 2020.
[17] Kynoch G. "Marashea on the mines: economic, social and criminal networks on the South African Gold Fields, 1947-1999." *Journal of Southern African Studies*, 26(1), 79-103, 2000.
[18] Xu T, Chen J, He Y, He DR. "Complex network properties of Chinese power grid". *International Journal of Modern Physics B*, 18(17-19), 2599-2603, 2004.
[19] Szklarczyk D, Morris JH, Cook H, Kuhn M, Wyder S, Simonovic M, et al. "The STRING database in 2017: quality-controlled protein-protein association networks, made broadly accessible". *Nucleic Acids Research*, 45, 362-368, 2017.
[20] Pal C, Acharyya A. *A novel architecture design for complex network measures of brain connectivity aiding diagnosis*. Editors: Garguilo GD, Naik GRG. Wearable/Personal Monitoring Devices Present to Future, 281-302, Singapore, Springer, 2022.
[21] Freeman LC. "Centrality in social networks conceptual clarification". *Social Networks*. 1(3), 215-39, 1978.
[22] Bonacich P, Lloyd P. "Eigenvector-like measures of centrality for asymmetric relations". *Social Networks*, 23(3), 191-201, 2001.
[23] Katz L. "A new status index derived from sociometric analysis". *Psychometrika*, 18(1), 39-43, 1953.
[24] Salehi A, Masoumi B. "KATZ centrality with biogeography-based optimization for influence maximization problem". *Journal of Combinatorial Optimization*, 40(1), 205-26, 2020.
[25] Rehm H, Matar M, Rombach P, McIntyre L. "The effect of the Katz parameter on node ranking, with a medical application". *Social Network Analysis and Mining,* 13, 1-8 2023.
[26] Goh KI, Kahng B, Kim D. "Universal Behavior of Load Distribution in Scale-Free Networks". *Physical Review Letters*, 87(27), 1-4, 2001.
[27] Boldi P, Vigna S. "Axioms for Centrality". *Internet Mathematics*, 10(3-4), 222-262, 2013.
[28] Qi X, Fuller E, Wu Q, Wu Y, Zhang CQ. "Laplacian centrality: A new centrality measure for weighted networks". *Information Sciences*, 194, 240-253, 2012.
[29] Zhang JX, Chen DB, Dong Q, Zhao ZD. "Identifying a set of influential spreaders in complex networks". *Scientific Reports*, 6(1), 1-10, 2016.
[30] Ma N, Guan J, Zhao, Y. "Bringing PageRank to the citation analysis". *Information Processing & Management*, 44(2), 800-810, 2008.
[31] Langville AN, Meyer CD. "A survey of eigenvector methods for web information retrieval". *SIAM review*, 47(1), 135-161, 2005.
[32] Tunali V, Tüysüz MAA. "Analysis of function-call graphs of open-source software systems using complex network analysis". *Pamukkale University Journal of Engineering Sciences*, 26(2), 352-358, 2020.
[33] Li Q, Zhou T, Lü L, Chen D. "Identifying influential spreaders by weighted LeaderRank". *Physica A: Statistical Mechanics and its Applications*, 404, 47-55, 2014.
[34] He Q, Lei Z, Wang X, Huang M, Cai Y. "An effective scheme to address influence maximization for opinion formation in social networks". *Transactions on Emerging Telecommunications Technologies*, 30(6), 1-15, 2019.
[35] Fei L, Zhang Q, Deng Y. "Identifying influential nodes in complex networks based on the inverse-square law". *Physica A: Statistical Mechanics and its Applications*, 512, 1044-1059, 2018.
[36] Wang Y, Li H, Zhang L, Zhao L, Li W. "Identifying influential nodes in social networks: Centripetal centrality and seed exclusion approach". *Chaos, Solitons & Fractals*, 162, 1-15, 2022.

[37] Pu J, Chen X, Wei D, Liu Q, Deng Y. "Identifying influential nodes based on local dimension". *Europhysics Letters*, 107(1), 1-6, 2014.
[38] Huang M, Zou G, Zhang B, Gan Y, Jiang S, Jiang K. "Identifying influential individuals in microblogging networks using graph partitioning". *Expert Systems with Applications*, 102, 70-82, 2018.
[39] Shang Q, Deng Y, Cheong KH. "Identifying influential nodes in complex networks: Effective distance gravity model". *Information Sciences*, 577, 162-179, 2021.
[40] Curado M, Tortosa L, Vicent J. F. "A novel measure to identify influential nodes: return random walk gravity centrality". *Information Sciences*, 628, 177-195, 2023.
[41] Xu G, Dong C. "CAGM: A communicability-based adaptive gravity model for influential nodes identification in complex networks". *Expert Systems with Applications*, 235, 1-15, 2024.
[42] Venunath M, Sujatha P, Koti P. "Identification of influential users in social media network using golden ratio optimization method". *Soft Computing*, 28(3), 2207-2222, 2024.
[43] Jiang C, Liu X, Zhang J, Yu X. "Compact models for influential nodes identification problem in directed networks". *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 30(5), 1-12, 2020.
[44] Davis TA, Hu Y. "The University of Florida Sparse Matrix Collection". *ACM Transactions on Mathematical Software (TOMS)*, 38(1), 1-25, 2011.
[45] Transportation Networks for Research Core Team. "Transportation Networks for Research". https://github.com/bstabler/TransportationNetworks. (05.10.2023).