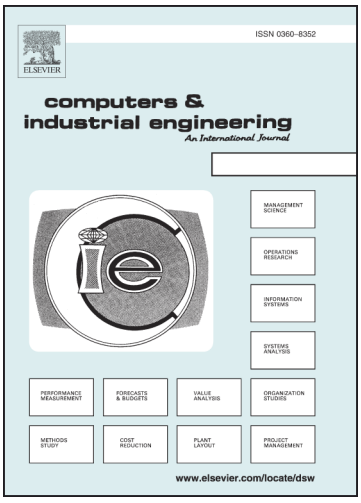


A hybrid algorithm based on community detection and multi attribute decision making for influence maximization

Masoud Jalayer, Morvarid Azheian, Mehrdad Agha Mohammad Ali Kermani

PII: S0360-8352(18)30193-1
DOI: https://doi.org/10.1016/j.cie.2018.04.049
Reference: CAIE 5198

To appear in: *Computers & Industrial Engineering*

Received Date: 29 July 2017
Revised Date: 3 March 2018
Accepted Date: 25 April 2018

Please cite this article as: Jalayer, M., Azheian, M., Agha Mohammad Ali Kermani, M., A hybrid algorithm based on community detection and multi attribute decision making for influence maximization, *Computers & Industrial Engineering* (2018), doi: https://doi.org/10.1016/j.cie.2018.04.049

A hybrid algorithm based on community detection and multi attribute decision making for influence maximization

Masoud Jalayer

Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Milan, Italy

Morvarid Azheian

Department of Progress Engineering, Iran University of Science and Technology, Tehran, Iran

Mehrdad Agha Mohammad Ali Kermani*

Department of Progress Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract

Influence maximization problem is trying to identify a set of *K* nodes by which the spread of influence, diseases or information is maximized. The optimization of influence by finding such a set is *NP-hard* problem and a key issue in analyzing complex networks. In this paper, a new greedy and hybrid approach based on a community detection algorithm and an MADM technique (TOPSIS) is proposed to cope with the problem, called, 'Greedy TOPSIS and Community-Based' (GTaCB) algorithm. The paper concisely introduces community detection and TOPSIS technique, then it presents the pseudo-code of the proposed algorithm. Afterwards, it compares the performance of the solution which found by GTaCB with some well-known greedy algorithms, based on *Degree Centrality*, *Closeness Centrality*, *Betweenness Centrality*, *PageRank* as well as TOPSIS, from two aspects: diffusion quality and diffusion speed. In order to evaluate the performance of GTaCB, computational experiments on eight different types of real-world networks are provided. The tests are conducted via one of the renowned epidemic diffusion models, namely, Susceptible-Infected-Recovered (SIR) model. The simulations exhibit that in most of the cases the proposed algorithm significantly outperforms the others, chiefly as number of initial nodes or probability of infection increases.

Keywords: Influence Maximization; Social Network Analysis; Community Detection; SIR model

* - Department of Progress Engineering, Iran University of Science and Technology, Farjam St, Narmak, Tehran, Iran , m_kermani@ iust.ac.ir, +989126850899

1. Introduction

Social networks permeate our social and economic lives. They play a central role in the transmission of information about job opportunities, and are critical to the trade of many goods and services[1]. Such networks also underlie the trade and exchange of goods in non-centralized markets, the provision of mutual insurance in developing countries, research and development, and collusive alliances among corporations, international alliances, and trading agreements; to mention just a few examples[2]. Social network analysis focuses on the relationship analysis between social and economic entities with the aim of finding common features between them[3, 4].

One of the most important and attractive research lines in social network analysis, is the analysis of information diffusion in social networks. Diffusion over social networks is a quite common phenomenon, like the spread of rumors, viral marketing of new products, virus propagation and public opinion formation[5-9]. Information diffusion is a vast research domain and has expressed research interests of many fields, such as Physics, Biology, etc.[10, 11]. For example, considering Facebook, where a user Sally updates her status or writes on a friend's wall about a new show in town that she enjoyed. The information concerning this action is typically passed on to her friends. When some of Sally's friends make comments on her update, the fact should not be forgotten that the information was flown to the friends of hers and the other users. In this way, the information provided by Sally has the potential to propagate transitively through the network[12].

One of the focal research directions related to information diffusion, is to conduct a study about how to choose individuals (here we call them 'seeds') to start the diffusion like that. When the diffusion process terminates, the number of infected individuals in the network can be maximized[5]. This problem would be of great importance to many companies as well as individuals that want to do a special promotion of their products, services, and innovative ideas through the powerful word-of-mouth effect (or called viral marketing)[13]. The above problem, called *influence maximization*, was first formulated as a discrete optimization problem in[14]: A social network is modeled as a graph with nodes representing individuals and edges as connections or relationships between two nodes. Influences are propagated in the network according to an influence model. Given a social network, an influence model and a small number as k, and the *influence maximization* problem is to find k nodes in the graph such that under the given influence model, the expected number of nodes activated by the k seeds is the largest possible [15].

This problem is generally classified into two separated classes[16]; competitive class and non-competitive class. In The first class, there are at least two decision makers desiring to spread their information onto a given network and attract as more nodes as possible to their favorite attitudes (such as[16, 17]). In the non-competitive case, however, only a single decision maker is present (such as[18]).

In this work, the focus is on the non-competitive case in which there is a decision maker attempting to spread a piece of information on a social network as much as possible. In this paper, a new algorithm to solve the influence maximization problem is proposed with three main features: 1) The proposed algorithm takes the topological features of the nodes into account. 2) The nodes' features are taking into consideration simultaneously by making use of one of the

multi attribute decision making methods. 3) The fact that the social networks hold some communities is strictly true.

So, the proposed algorithm in the present paper is trying to find the most influential nodes in a given network based on multiple criteria (multiple centrality measures). In the existing works, a number of centrality measures have been proposed to identify the influential nodes[19, 20]. However, all of them focused on only one centrality measure and they have some limitations and disadvantages[21]. Meng *et al.* showed that the centrality measures have different performance to find the influential nodes[22]. If only one centrality measure is adopted, then the rankings of identifying the influential nodes may be different by using a different centrality measure[23]. So, to cope with this inefficiency in finding the most influential nodes in social networks, one of the multi attribute decision making techniques (TOPSIS) is utilized in the present work.

On the other hand, since the social networks have a feature that they are community-based[24], we come to the conclusion that by exploring the community structures naturally embedded in a social network, efficient algorithms can be developed to address the influence maximization problem[25].

Since there are no algorithms in which a MADM method and community detection algorithm are utilized in the influence maximization simultaneously, the main novelty of the present work is to consider the multiple centrality measures to find the most influential nodes in each community to find the best seed nodes to maximize the influence on the social networks.

The rest of the paper is organized as follow: The next section deals with the literature review of influence maximization problem. The third section of the paper defines the considered problem. The suggested algorithm and the used methods are elaborated in the fourth one. The following part expresses the efficiency of the proposed algorithm using some real well-known datasets in the literature.

2. Literature Review

The influence maximization problem has been proposed and studied by Domingos and Richardson in 2001 [26] and was followed by Kempe *et al*. in 2003[14]. This problem can be mathematically defined on a network $G = (V, E)$, under one of the influence models such as independent cascade or linear threshold. Let n and u be the numbers of elements of V and E, respectively. Let K be a positive integer with $n > K$. So, the influence maximization problem is finding a set A_K^* of K nodes to target for initial activation such that $\sigma(A_K^*) \geq \sigma(S)$ for any set S of nodes, i.e.

$$A_K^* = \underset{A \epsilon \{S \subset V; |S| = K\}}{\operatorname{argmax}} \sigma(A)$$

Where $|S|$ stands for the number of elements of set S and $\sigma(S)$ stands the number of infected nodes in termination of information diffusion if S is the set of initial nodes for information diffusion process.

In this section, at first, some of the well-known influence models used in previous works are reviewed. The next subsection reviews some of the classic works. The third part of this section is dealing with some of works which investigate the problem from a multi attribute point of view. And finally, the papers in which a community-based approach is adopted to cope the problem are reviewed.

2.1. Influence models

Let the diffusion of information on social networks proceeds along discrete time phases ($t = 0,1,2,\ldots$). Each node ($v \epsilon V$) can be either active (infected) or inactive (non-infected). The influence models try to describe the status of the nodes based on different initial nodes sets in each step. In this subsection, we review four main and well-known influence models, named, Independent Cascade (IC), Linear Threshold (LT), Susceptible-Infected-Recovered (SIR) and Susceptible-Infected-Susceptible (SIS). It should be noted that IC and LT are known as stochastic diffusion models and SIS and SIR are known as epidemic model in the literature[12].

The IC model considers the network as $G = (V, E)$, the influential probability $p(.)$ on all arcs, and an initial seed set S_0 as the model's inputs. Then it will generate the active set of nodes in each time phase (S_t) by the following rule. In each step ($t \geq 1$), the first set S_t to be S_{t-1}; next for every inactive node $v \notin S_{t-1}$, for every node $u \notin N^{in}(v) \cap (S_{t-1} \backslash S_{t-2})$, u executes an activation attempt by performing a Bernoulli trial with in a probability of success $p(u, v)$; if successful we add v to S_t and v is activated by u at time t.

The LT model considers the network as $G = (V, E)$, the influence probability $w(.)$ on all arcs, and an initial seed set S_0 as the model's inputs. Then it will generate the active set of nodes in each time step (S_t) by the following rule. In $t = 0$, each node ($v \epsilon V$) independently selects a number at random (θ_v) from a uniform distribution in the range $[0,1]$. In each step ($t \geq 1$), first set S_t to be S_{t-1}; then for any inactive nodes $v \epsilon V \backslash S_{t-1}$, if the total weight of the arcs is at least θ_v from its active in-neighbors, then v will be activated and added to S_t.

SIR model of infection is one of the most renowned infection models widely used in social network simulation and analysis, it is also mapped onto a bond percolation model [27, 28],in which each individual (or node) transitions between several possible states, which typically include state *S* (for *susceptible*), state *I* (for *infected*), and state *R* (for *recovered* or *removed*). A node in state *S* has not the disease but is susceptible to get the disease upon contact with an infected node. A node in state *I* has the disease and can transmit the disease to susceptible nodes upon contact, with infection rate β, which is interpreted as the probability of successful transmission of the disease from an infected node to a susceptible node in a time unit. Consider the network of relationship as $G = (V, E)$, where S_t is the set of susceptible nodes, I_t denotes the set of Infected nodes and R_t, the set of Recovered ones at time *t*; here if $v \in S_t$, it becomes infected in *t+1* if it has at least one neighbor $u \in I_t$, who successes to spread the infection on v, with Bernoulli success probability of $p = w_{v,u} \times \alpha$; in which $w_{u,v} \in [0,1]$ represents the weight of arc connecting u to v where $\alpha \in [0,1]$ is the infectiousness rate of u at time *t*. In such a way $\begin{cases} v \in I_{t'}, \ t \leq t' \leq t + L \\ v \in R_{t'}, \ t' > t + 1 + L \end{cases}$, so the process ceases whenever the network experience an steady state where all infected nodes are recovered[29].

2.2. Classical influence maximization

Kempe *et al*. proved that the influence maximization problem is *NP-hard*, and proposed a greedy approximate algorithm considering *LT*, *IC* and *WIC*[†], which guarantees that the influence spread is within $(1 - 1/e)$ of the optimal solution[14]. They also showed in experiments that their greedy algorithm significantly outperformed the classic degree and centrality-based heuristics in

[†] - Weighted Independent Cascade

influence spread. Afterwards, there were many studies which proposed different algorithms to find the best set of initial nodes with influence spread.

For example, Leskovec et al. [30]proposed an improved greedy algorithm by introducing a "Cost-Efficient Lazy Forward" (CELF) scheme. The CELF algorithm can speed up the greedy algorithm by 700 times. Following the kempe *et al*'s work[14], Goyal et al.[31] optimized CELF by exploiting sub-modularity and experiments and proposed CELF^{++} algorithm. They showed that CELF^{++} algorithm is 35% ~55% faster than CELF. In another research, Chen *et al*.[13] developed the New-Greedy and Mixed-Greedy algorithms to improve the greedy algorithm in different ways. Liu et al. proposed a bound linear approach to influence computation and influence maximization[32]. Kermani *et al*.[11] proposed a multi objective mathematical program with linear objectives and constraints in search of the seed nodes in social networks. Since the proposed model was solved by an exact algorithm (CPLEX), they claimed that their model finds the optimal solution for influence maximization problem.

Additionally, some heuristic algorithms have been proposed to cope with the influence maximization problem. These approaches are trying to find the top-k nodes in social networks based on degrees or other centrality measures. Recently, some studies have been made in which some meta-heuristic based algorithms were proposed to deal with the influence maximization problem. For example, Yang and Weng[33] proposed the swarm intelligence-based algorithm (ant-colony optimization algorithm) to consider the influence maximization problem. The proposed algorithm was evaluated using a co-authorship data set and the obtained experimental results showed that the proposed algorithm outperforms two well-known benchmark heuristics. Other metaheuristic algorithms such as genetic algorithm[34], simulated annealing algorithm[35], particle swarm optimization algorithm[36] and cuckoo search algorithm[37] have been utilized to deal with the influence maximization problem, too.

2.3. Multi-attribute-based approach for influence maximization

Another wave in influence maximization research line is to find the influential nodes (key nodes) in consideration of more than one criterion. For the first time, Mesgari *et al.* in 2013, presented a novel approach in a conference in Bielefeld, in which they utilized TOPSIS method to find the key nodes in social networks[38]. They examined their approach using three datasets in various sizes and sub-structures. In another study, Fox and Everton applied a hybrid method based on AHP and TOPSIS to find the influential nodes in Noordin Dark Network. Additionally, they discussed a bit about the sensitivity of the nodes' rank based on changes in weights of the criteria[39]. Zhang *et al.* studied the problem of nodes' importance in the research and the developmental team[40]. They considered the eight criteria to identify the importance of the nodes; there were four criteria from centrality measures (degree centrality, betweenness centrality, closeness centrality, and eigenvector centrality) and the others from structural holes of complex networks (effective size, efficiency, constraint, and hierarchy). They used a Fuzzy AHP method to identify the weights of the mentioned criteria and then a TOPSIS method to rank the nodes based on their importance. Du et al. proposed a TOPSIS method to identify the influential nodes in social networks[23]. They applied different types of centrality measures as TOPSIS's attributes in different networks. The effectiveness of the proposed method considering SIS model as the influence model, is examined by comparing the results with some of the benchmark methods. The main weakness of Du et al.'s work was the consideration of the same weights for TOPSIS's attributes in nodes rank. The proposed method has improved in[23]. The authors proposed a new algorithm to calculate the weight of each attribute. In order to evaluate the

performance of the method, they used the SIS model as the influence model to simulate the diffusion process in four real networks.

2.4. Community-based approaches for influence maximization

Another wave in influence maximization research line is developing community-based approaches to solve the problem. Community structure is defined as the division of network nodes into groups, within which nodes are densely connected while between which they are sparsely connected[41]. The main reason of utilizing community-based approach to this problem is the reduction of the computational time and an increase in the performance.

Cao *et al.* proposed the first community-based influence maximization algorithm OASNET (Optimal Allocation in a Social NETwork). They transformed the influence maximization problem into an optimal resource allocation problem. Also, they assumed that different communities are independent of each other and influence cannot spread across different communities[42] .Then, they proposed a recursive relation to find the influential nodes in social networks. Zhang *et al.* studied the problem of influence maximization on networks with community structure. The authors constructed an information transfer probability matrix from the weighted network[43]. Then they applied the k-medoid clustering algorithm to identify the Top-*K* (influential) nodes. The performance of the proposed method has been investigated using LFR synthetic networks[44] and several real-world network. Wang *et al.* proposed an algorithm called Community-based Greedy Algorithm for mining top-K influential nodes in social networks[45]. Their empirical studies show that their method is faster than the state-of-the-art Greedy algorithm to find the influential nodes. Chen *et al.*[25] developed a new framework to tackle the influence maximization problem with an emphasis on time efficiency. The proposed framework consists of three phases; *community detection*, *candidate generation* and *seed selection*. They tested the proposed framework's efficiency and scalability on both synthetic and real datasets and showed the developed framework outperforms the state-of-the-art algorithms. One of the most recent studies in this field is Shang *et al.*'s research[18]. To solve the influence maximization problem, they proposed a new algorithm named as *CoFIM*. The developed algorithm contains two phases; *seeds expansion* and *intra-community propagation*. The first phase is the expansion of seed nodes among different communities at the beginning of the diffusion. The second phase is the influence propagation within communities which are independent of each other. To evaluate the performance of the proposed algorithm with state-of-the-art algorithms, they used some synthetic and real-world large datasets.

3. Problem Definition

We consider a directed social network $G = (V, E)$, with $|V| = n$ and $|E| = u$ with the edge weights w_{ij} between nodes *i* and *j*. In this paper, we assume that the influence model is SIR and the time proceeds in discrete periods. Furthermore, based on the mainstream of the influence maximization literature, it is assumed that the model is progressive. We assume at time t = 0, the seed nodes are active. An active node stays infectious for L periods when she can infect its immediate neighbors with given infectiousness rates, and be '*Recovered*' after that, so that she doesn't have a chance to infect others any more. Hence, the process lasts, when there is at least one active node, and terminates immediately when the last active nodes are Recovered. The focus of this paper is to find a set of seeds maximizing the number of infected nodes at the cessation of the infection process.

4. The Proposed Algorithm

In this paper, a new algorithm is proposed to cope with the influence maximization problem given the SIR influence model. The proposed algorithm is called "*Greedy TOPSIS and Community Based"* algorithm which is abbreviated to *GTaCB*. The algorithm employs two well-known methods in community detection literature and multi-attribute decision making. The procedure utilized the methods and the result of the implementation of the proposed algorithm on a small dataset being introduced in ***4.1.***

4.1. Procedure

The schematic procedure of the proposed algorithm is illustrated in ***Fig.1***. First, *GTaCB* runs a community detection algorithm to find *K* partitions within the graph, which is explained in 4.1.1. In some cases, however, it is possible that the graph doesn't contain *K* distinct communities with respect to the community detection algorithm used. Therefore, given it detects *H* communities, *GTaCB* divides the whole graph, into *H* sub-graphs with respect to the communities. Afterwards, in each sub-graph the Centrality Analysis and then TOPSIS technique has to be conducted, which is explained in ***4.1.2***. Finally, to select the set of seed nodes as the output of the algorithm, it sorts the sub-graphs by their number of nodes, decreasingly. Then, as a loop, it starts from the premier sub-graphs and allocates the highest ranked node of each one to the set of seeds, *S*, and removes the allocated nodes from the rankings. In the case that *K*>*H*, for the remaining desired seeds, the algorithm uses the next ranked nodes in each iteration until it is satisfied.

Fig.1 – The schematic process of GTaCB algorithm

4.1.1. Community Detection

Detecting the communities in networks is a big challenge for which many methods and algorithms have been proposed in the last decades, within different scientific disciplines such as Physics, Biology, Computer Science and Social Sciences[46]. There are different algorithms identifying the communities each of which are used for different types of networks, depending on the network features and the characteristics of the community detection algorithm[47, 48].

Since the proposed algorithm in this paper is trying to find *k* best influential nodes in *k* communities of social networks, it is preferred to use a community detection algorithm in which the number of communities can be tuned before running the algorithm. Graph Community detection by Spectral Clustering (GCSC) algorithm proposed by Hespanha[49] is a community detection in which the number of desired communities is one of its inputs. So, we implement this algorithm to divide the graph into the partitions in a way that minimizes the edge-costs of each partition.

Let $G = (V, E)$ represent a directed graph, where V is the set of vertices with edge set of E, and there are k-partitions (subsets) of V denoted as $P = \{V_1, V_2, \dots, V_k\}$ where $V_i \cap V_j = \emptyset \,; i \neq j \in \{1, \dots, k\}$ and $V_1 \cup V_2 \cup, \dots, \cup V_k = V$

An edge-costs function of graph partitioning P can be defined as:

$$C(P) = \sum_{i \neq j} \sum_{\substack{(v,\bar{v}) \in E \\ v \in V_i, \bar{v} \in V_j}} c(v, \bar{v}) \;\; ; \; c : E \to [0, \infty)$$

4.1.2. Multi Attribute Decision Making

As it's illustrated at ***Fig.1***, the process of identifying the influential nodes is initiated by the community detection algorithm and then if the desired number of partitions is found, the processor inserts the detected partitions into the algorithm calculating centrality measures. Then TOPSIS (or any other MADM tool) helps us select the top scored node of each partition.

Hwang and Yoon [50] introduced TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution) which has become one of the most prevalent MADM (Multi-Attribute Decision Making) methods in the literature. In such a method, there is a finite set of alternatives about to be evaluated and ranked by the criteria or attributes which are individually weighted before. TOPSIS has been utilized in many research areas such as Supply Chain Management[51], Facility Location[52], HSE[53] and Project Management[54]. In our algorithm, we defined the criteria as network centrality measures. TOPSIS suggests herewith the alternative which is the closest to the ideal solution and the farthest from negative ideal solution as the most influential node in each community [55]. TOPSIS has following steps[38]:

- **Step1**: Creating a decision matrix with *m* rows as alternatives (nodes) and *n* columns for criteria: $X = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mp} \end{bmatrix}$.
- **Step2**: The normalization step in which *X* is converted to *R* by $r_{ij} = x_{ij}\left(\sum_{i=1}^{m} x_{ij}^2\right)^{-1/2} \; \forall j$
- **Step3**: Determining the weight normalized matrix $\left[t_{ij}\right]_{m\times p} = \left[w_j \, r_{ij}\right]_{m\times p}$ showing the relative importance of each criterion.
- **Step4**: Let J_+ represent the set of benefit criteria and J_- as the set of cost criteria, determine the positive ideal solution (t_j^+) and the negative ideal solution (t_j^-) as:
  - $t_j^+ = \left\{\left(Max\; t_{ij} \middle| i = 1, \dots, m\;;\; \forall j \in J_+\right), \left(Min\; t_{ij} \middle| i = 1, \dots, m\;;\; \forall j \in J_-\right)\right\}$
  - $t_j^- = \left\{\left(Max\; t_{ij} \middle| i = 1, \dots, m\;;\; \forall j \in J_-\right), \left(Min\; t_{ij} \middle| i = 1, \dots, m\;;\; \forall j \in J_+\right)\right\}$
- **Step5**: Calculating the distance of each alternative from its positive and negative ideal solution by: $S_i^+ = \sqrt{\sum_{j=1}^{p}\left(t_{ij} - t_j^+\right)^2}\;;\; S_i^- = \sqrt{\sum_{j=1}^{p}\left(t_{ij} - t_j^-\right)^2}\;; i = 1, \dots, m$
- **Step6**: Determining the relative closeness to the ideal solution for all nodes, as: $C_i^* = \frac{S_i^-}{S_i^- + S_i^+}$
- **Step7**: At the final step, the nodes are ranked based on C_i^* values, in descending order.

Based on the previous works using TOPSIS methods to identify the most influential nodes in social networks, we considered *Degree Centrality (DC), Closeness Centrality (CC), Betweenness Centrality (BC) and PageRank (PR)* as the attributes in this decision-making process[23, 38, 56].

4.1.3. The GTaCB algorithm

So, the identified influential set of seeds consist of all highest ranked nodes of *k*-partitions (*P*). in the remainder of the section, firstly, we define the main notations used in the paper in ***Table 1***, secondly, the pseudo-code of the proposed algorithm is exhibited in ***Algorithm 1*** and then the paper gives an example on a small network and compares the proposed algorithm results with those of TOPSIS.

Table 1 – Main Notations

Parameter	*Definition*
n	Number of nodes in the network
u	Number of edges between nodes in the network
V	Set of n nodes
E	Set of u edges
w_{ij}	Weight of the edge connecting *i* to *j*
K	Number of initial seeds
S^t	Set of susceptible nodes at time *t*
I^t	Set of infected nodes at time *t*
R^t	Set of recovered nodes at time *t*
p_{ij}^t	Probability of transmission the infection/influence from *i* to *j*, if $i \in I^t$ and $j \in S^t$
L	Number of periods a node stays infected
α_r	Infectiousness rate in r^{th} period of infection; $r = 1,\ldots,L$
k	Relative infectiousness
Γ	Number of infected nodes at the end of the infection process
τ	Length of the infection process (Number of periods the process lasts)
η	Diffusion speed

With using the notations described in ***Table 1***, the paper proposes GTaCB as follow:

Algorithm 1 – GTaCB

Initialization: *V; E; G { V , E } = directed graph;*
Parameter Settings: *K; W = vector of relative weights for TOPSIS attributes*
Main Steps:
S = zeros*(1,K); % an empty vector as a container for initial seeds*
Community Detection: *P = GCSC(G,K); % returns P as the set of K Partitions*
Sorting the Communities: *sort vector P w.r.t the number of nodes, decreasingly.*
Seed Detection:
for *p = 1: length(P)*
  P_P = the subgraph of G, containing vertices in p^{th} subset of P;
  Centrality Measures Calculation: *C = Centralities(P_P);*
  % returns the matrix of C containing each node's centrality values in subgraph P_P
  TOPSIS Calculation: *T = TOPSIS(C,W);*
  % returns nodes of P_P as the vector T in descending order by their TOPSIS ranks
  if *p<= K-floor(K/length(P))*length(P)*
    S = [S, T(1, 1 : ceil(K/length(P))];
    % allocates the best ranked nodes to the Seeds
  else
    S = [S, T(1, 1 : floor(K/length(P))];
    % allocates the best ranked nodes to the Seeds
  end
end

4.2. Example Explanation

Let us consider a small synthetic undirected network named 'Ex1' having 20 nodes, with equal weight values on its edges. The network has been generated by 'Random Modular Graph[‡]' algorithm programmed by MIT Strategic Engineering Research Group, the inputs of which are n=20, c=2, p=0.3 and r=0.9. The graph depicts in ***Fig.2*** with distinct colors for its communities.

Fig.2 – Ex1 with two communities

[‡] http://strategic.mit.edu/downloads.php?page=matlab_networks

Consider if there are only two seeds to start the infection with. According to their centrality measures (DC, CC, BC and PR), ***Table 2*** contains the values each node has in the network. Hence, the top two nodes of each centrality's ranking are selected and highlighted in the table.

Table 2 – The ranks of the nodes by each centrality (highlighted rows are selected)

Rank	BC		CC		DC		PR	
	NODE	SCORE	NODE	SCORE	NODE	SCORE	NODE	SCORE
1	**18**	**57.55433**	**18**	**0.032258**	**18**	**9**	**18**	**0.07376**
2	**5**	**34.1868**	**2**	**0.03125**	**11**	**8**	**13**	**0.068025**
3	11	32.39545	6	0.030303	13	8	11	0.065967
4	6	32.22749	9	0.030303	2	7	5	0.058792
5	8	27.24242	10	0.030303	3	7	6	0.057705
6	17	25.7868	11	0.030303	5	7	3	0.057703
7	10	24.51558	17	0.030303	6	7	8	0.057076
8	13	23.48831	8	0.029412	8	7	2	0.05656
9	2	21.3026	5	0.028571	10	7	10	0.056332
10	12	18.68918	13	0.028571	7	6	17	0.050703
11	9	18.33593	12	0.027778	9	6	9	0.049483
12	3	12.13355	3	0.027027	17	6	7	0.04941
13	20	11.72641	7	0.025641	1	5	15	0.044682
14	7	4.307143	19	0.025641	12	5	12	0.04442
15	15	4.200866	20	0.025641	15	5	19	0.043488
16	1	2.383333	15	0.02439	19	5	1	0.042314
17	19	1.357143	16	0.02439	16	4	16	0.036251
18	14	1.083333	1	0.02381	4	3	20	0.029452
19	16	1.083333	14	0.023256	14	3	14	0.029237
20	4	0	4	0.022222	20	3	4	0.028641

According to the steps of the proposed algorithm, the communities are detected and shown in ***Table 3***, so the graph is split into two sub-graphs, in each of which the centralities and subsequently the TOPSIS scores are calculated independently. So, in community #1, node 3 is chosen and in community #2, node 13. However, nodes 18 and 11 would be selected as the first and the second ones ranked as if the graph were not separated.

Table 3 – The ranks of the nodes via TOPSIS and GTaCB (highlighted rows are selected)

Rank	**TOPSIS**		**Rank**	**GTaCB**		
	NODE	SCORE		NODE	Scores	Community
1	**18**	**1**	**1**	**3**	**1**	1
2	**11**	**0.617439**	**1**	**13**	**1**	2
3	5	0.609788	**3**	5	0.880983	1
4	6	0.583636	**4**	6	0.571812	1
5	8	0.51454	**5**	11	0.530511	2
6	13	0.505129	**6**	18	0.511357	2
7	10	0.479036	**7**	7	0.486803	1
8	17	0.464143	**8**	10	0.470492	1
9	2	0.441045	**9**	8	0.461369	1

10	9	0.365922	**10**	2	0.446429	1
11	3	0.335177	**11**	9	0.30307	1
12	12	0.332455	**12**	1	0.302609	1
13	7	0.217104	**13**	15	0.284238	2
14	20	0.179894	**14**	17	0.220245	2
15	15	0.16514	**14**	19	0.220245	2
16	19	0.148921	**16**	12	0.167151	2
17	1	0.144904	**17**	16	0.161097	2
18	16	0.081011	**18**	14	0.093239	2
19	14	0.021391	**19**	4	0	1
20	4	0	**19**	20	0	2

Fig.3 exhibits the results of an iterative simulation on Ex1 for all selected seeds which are inputted into an SIR model, where maximum iteration is 2000, $L = 2$ and $\alpha = (0.3,0.15)$ and the probability of infection is $p_{i,j} = k\alpha w_{i,j}$.

The results shown in ***Fig.3*** and ***Fig.4*** show that the seeds that GTaCB suggests, infect more nodes through the iterations, particularly than TOPSIS.

Fig.3 – Ex1 SIR infection results with 2 seeds

Fig.4 - Ex1 SIR infection results with 4 seeds

5. Experimental Results

In order to evaluate the performance of GTaCB algorithm in comparison with some other famous approaches, we have simulated the spread of influence of chosen initial nodes detected by each measure, using an SIR model[41], an epidemic model of infection. All the codes have been written in MATLAB R2016a, and the results are computed at a Windows 8.1 OS with Core i7 Intel CPU of 3.1GHz and 8GB memory.

5.1. Real-World Datasets

To achieve a kind of comprehensive comparable result, it is worth employing an adequate variety of datasets exhibiting satisfactory structural features of real networks[14]. Therefore, in this experiment, there are 9 different real-world networks on which we have tested the spread of selected seeds' influence. These datasets are introduced as follows:

- Abrar: A network of SMS connections between university students in industrial engineering and computer engineering at a higher education institute in Tehran named 'Abrar', between the years 2010 and 2011[57].
- USAir: The network of North American Transportation Atlas Data (NORTAD) contains geographic data sets for transportation facilities in Canada, Mexico, and the United States. http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm
- EuroSiS: The network based on collaborations between "European Science in Society" and agents, realized in a WebAtlas study among 12 Countries in Europe, which is accessible at: https://github.com/gephi/gephi/wiki/Datasets

- OCLinks: An online weighted social network created based on students' online message interactions through an online community at the University of California, Irvine. The weight of an edge is defined as the number of messages sent over a period from April to October 2004. https://github.com/gephi/gephi/wiki/Datasets
- Yeast: A dataset of unweighted networks representing protein-protein interactions in budding yeast based on an innovative interactive detective study done by Dongbo Bu et al. *[43]*.
- Geom: A weighted graph obtained from computational geometry collaborations among authors who had any jointly publishing work. The weights representing number of works each pair of nodes co-authorized published. http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm , https://arxiv.org/
- HEP-th: Also, is known as "High Energy Collaboration", is a weighted indirect graph illustrating the posting preprints between physicists in the field of "High Energy Physics" theory E-Print Archive from the beginning of January, 1995 until the last day of 20th century. The graph datasets are accessible at http://www-personal.umich.edu/~mejn/netdata/ also the comprehensive E-Print archive data is open accessed at the Cornell University Library at: https://arxiv.org/
- PGP-Giant: The network of users who shared confidential information via an encryption algorithm called Pretty-Good-Privacy. These interactions made an edg list of the giant component of this graph in 2004 by Boguñá et al.[58] and the dataset is available at: http://deim.urv.cat/~alexandre.arenas/data/welcome.htm
- Slashdot: A news website which features user-submitted and editor-evaluated currents of primarily technology-oriented news. After 2002, it allows users to tag each other as friends or foes. The network contains friend/foe links between the users of Slashdot in Feb. 2009. The dataset is available at: http://snap.stanford.edu/data/soc-sign-Slashdot090221.html

The properties of these real-world datasets are introduced in ***Table 4***:

Table 4: Properties of real-world datasets

	Abrar Ins.	**US Airlines**	**EuroSiS**	**OCLinks**	**Yeast**	**Geom**	**HEP-th**	**PGP-Giant**	**Slashdot**
Network Type	Friendship	Transportation	Collaboration	Online Social Network	Biology	Collaboration	Co-authorship	Information Sharing	Friendship
#Nodes	163	332	1285	1899	2361	7343	8361	10680	82140
#Edges (Directed)	3113	2126	7524	20296	14364	23796	31502	48632	549202
#Strongly Connected Components	1	332	511	601	2361	7343	8361	1	1
Network Diameter	5	6	14	8	16	12	17	24	12
Average Path Length	2.466	2.563	4.943	3.197	4.647	4.006	5.16	7.485	10.17
Average Degree	38.196	12.807	11.711	21.375	6.084	3.241	3.768	9.107	13.372
Maximum Out-Degree	51	99	98	237	60	101	23	205	2548
Maximum Betweenness	2599.23	5286.21	162757.11	148225.36	36248.02	46776.07	25686.02	14959584.71	550761.31

Fig. 5 - The visualizations of Slashdot(a), PGP (b), HEP-th(c), Geom(d), Yeast(e), OCLinks(f), EuroSiS(g), USAir(h) and Abrar(i) graphs – The layouts are based on Force-Atlas algorithm

Firstly, we divided these graphs into *K* clusters via GCSC algorithm as it is explained in ***4.1.1***, for instance, ***Fig. 6*** depicts the different identified communities in PGP graph:

Fig. 6 – PGP graph with 50 (right) and 250 (left) detected communities. The colors represent different identified communities

If we denote the set of top *K* nodes which are ranked by i^{th} algorithm as S_i, Jaccard Coefficient (*JC*) between i^{th} and j^{th} algorithms' detected seeds can be defined as follows,[59]:

$$JC_{ij} = \frac{\|S_i \cap S_j\|}{\|S_i \cup S_j\|}$$

Accordingly, ***Fig. 7***, ***Fig. 8*** and ***Fig. 9*** show *JC* values between all pairs of detected seeds by every algorithm in the 3 networks respectively: PGP, GEOM, and Yeast. The figures demonstrate that GTaCB has the least *JC* in comparison with the others unlike those of TOPSIS's, whose *JC* is the highest in terms of centrality. The figures show that the output of TOPSIS has the commonest nodes with the utilized algorithms, however it would not be unanticipated due to the inherency of an MADM technique. It can be clearly seen in our tests that GTaCB's relationships with PR and BC were higher than its relationship with CC or the rests, especially as *K* increases. It also demonstrates that the majority of individual nodes, selected by the proposed algorithm, are not considered as "influential" by well-known centralities.

Fig. 7 – The relationship between each pair of utilized algorithms' detected seeds in PGP network

Fig. 8 – The relationship between each pair of utilized algorithms' detected seeds in GEOM network

Fig. 9 – The relationship between each pair of utilized algorithms' detected seeds in Yeast network

5.2. SIR model results

In order to achieve a quantitative analysis of the proposed algorithm, a modified SIR model of infection is utilized, to which Jaquet and Pechal [41] have imparted a new parameter named 'relative infectiousness' that multiplies in all values of infectiousness rate vector (α_i), it helps us extract a large spectrum of infection conditions by increasing this parameter from 0 to 1. The Pseudo-Code of this simulation is written in the ***Algorithm 2***.

Algorithm 2 – SIR Model Simulator

Initialization: *V; E;* let G be the directed graph;
Parameter Settings: *k;* α *= infectiousness rate vector; L;*
itermax = number of iterations; Seeds = set of initial seeds; K;
Main Steps:

```
infect N(Seeds) as initial seed;
Ψ = zeros(n,itermax);
for iter= 1: itermax
    Calculate the number of active infected nodes and put it into C_inf ; t = 1 ;
    while C_inf>0
        for i= 1: K
            J(i) = set of infected nodes which have a directed edge to node i;
            if i is susceptible and J(i)≠ϕ
                infect i with probability of p(i) = k Σ_j w_ij α_s ; j ∈ J(i)
            end
        end
        update the state of all nodes; update C_inf ;
        t = t+1;
    end
    γ = vertical vector of nodes at which infected nodes take 1, and others 0;
    Ψ (:, iter) = γ ;
end
Let ψ(Seeds) be a vector with average of Ψ matrix on iterations;
```

The top K ranked nodes of each algorithm ($Seeds_{i.K}$) is inputted to ***Algorithm 2*** as the initial nodes. Parameter settings were the same for all of our employed networks, where $L = 2$ and $\alpha = (0.30\ \ 0.15)$, and the number of iteration was varied between 100 and 500 in each network (since the process was too long for larger networks), it guarantees an evenhanded conclusion. All the results were stored and for this paper 1,445,000 times ***Algorithm 2*** has been called to simulate the result on the whole. For each case, we have collected two simulated outcomes:

- The average number of infected nodes through the iterations (Γ).
- The average number of periods it lasted to get a steady step in the iterations (τ).

5.2.1. Diffusion Quality of GTaCB

In these comparisons, an algorithm whose set of seeds infects a larger number of nodes at the end of diffusion periods, has a higher quality. Thus, in ***Fig. 10***, where $K \geq 50$, Γ values of PGP network illustrates that GTaCB outperforms other algorithms when k is greater than 0.1, in this condition, CC and DC's seeds infect less than others and PR and BC follow GTaCB alongside each other. For instance, when $k = 0.5\ and\ K = 200$, CC could infect near 2296 nodes at average, there were approximately 348 and 301 nodes less than those of BC (2644.88) and PR (2597.76), respectively, however GTaCB's set of seeds infect more than 2871 nodes, puts this algorithm in the first place. By comparing ***Fig.10*** and ***Fig.11***, it can clearly be seen that the gap between the algorithms' qualities opens up, as either K or k increases. So, in PGP network as $K \geq 50$ and $k > 0.1$, our algorithm significantly shows a higher performance in terms of diffusion quality. For instance, in our tests, when $K = 250$ and $k = 0.4$, GTaCB's seeds have infected 20.4%, 10.4%, 27.6%, 8.3% and 38.9% more than those of TOPSIS, PR, DC, BC and CC, respectively.

Fig. 10 – Infection result comparisons in PGP Network

Fig. 11 – Infection result comparisons in 3D-bar chart of PGP Network

Just alike PGP's result, the epidemic spreading results in Slashdot, HEP-th, Geom, Yeast, OCLinks, USAir and Abrar networks exhibit the domination of GTaCB performance, chiefly as K and k increase; **Fig. 12** to ***Fig. 18*** illustrate the experimental results of these networks respectively:

Fig. 12 – Infection result comparisons in Slashdot Network

Fig. 13 – Infection result comparisons in HEP-th Network

Fig. 14 – Infection result comparisons in GEOM Network

Fig. 15 – Infection result comparisons in YEAST network

Fig. 16 – Infection result comparisons in OCLinks Network

Fig. 17 – Infection result comparisons in USAir Network

Fig. 18 – Infection result comparisons in Abrar Network

Nevertheless, in our tests there was a network (EuroSiS) in which the quality of PR had dominion over its rivals. However, GTaCB was following it as the second highest quality in most of the situations can be seen in ***Fig. 19***.

Fig. 19 – Infection result comparisons in EuroSiS Network

By averaging $\Gamma_{i.K.k}$ values on K and k values, we summarized the results in ***Table 5***, to provide insights as to how GTaCB's quality of diffusion outperforms the comparable algorithms through the examinations on Abrar, USAir, OCLinks, HEP-th, GEOM and PGP. Notwithstanding its quality, the proposed approach has a drawback of its runtime, due to the community detection techniques dullness and memory usage so that, the implementation of larger networks was almost impossible on the foresaid laptop, because of the "Out of Memory" MATLAB error.

Table 5 – Mean of infected nodes percentages on K_r and k_m conducted by different algorithms on the employed networks.

$\overline{\Gamma}$	GTaCB	TOPSIS	PR	DC	BC	CC
Abrar	88.22%	83.82%	85.09%	82.88%	84.37%	83.27%
USAIR	37.57%	31.02%	31.23%	30.89%	32.22%	30.95%
EuroSiS	34.87%	33.80%	35.96%	31.43%	34.00%	31.16%

OCLinks	52.28%	49.61%	49.65%	49.60%	49.63%	49.57%
YEAST	42.16%	40.72%	41.01%	40.62%	40.89%	40.22%
Geom	24.32%	22.09%	22.25%	21.77%	22.32%	21.40%
HEP-th	19.37%	17.80%	18.32%	17.57%	17.90%	17.09%
PGP	24.01%	22.35%	22.59%	21.76%	22.77%	21.36%
Slashdot	19.22%	16.70%	17.19%	15.74%	17.50%	15.44%

5.2.2. Diffusion Speed of GTaCB

As mentioned above, all τ values have been collected, and in ***Fig. 20*** it is shown that as k grows, τ values of GTaCB decrease as it did in PGP, HEP-th, Yeast and Abrar. It spreads in a shorter time in comparison with those of its rivals.

Fig. 20 – The comparison of τ values for the employed networks

Its average τ value is almost more than others in many cases, as depicted in ***Table 6***. Since these values are depended on the diffusion quality, there is an ambiguity to judge which algorithms' set of seeds infect more nodes in quickly. For example, in USAir network, diffusion process of DC's set of seeds lasts 3.99 periods on average, which is the shortest time in comparison with those of others, but it infects only 30.89% of the network. While, GTaCB' set has the most average longest time of 5.91 periods, but it infects 37.57% which outnumber others. Hence, we've defined another simple but applicable variable showing that how many nodes are infected through each period averagely. It makes us find out a kind of diffusion speed measure to evaluate the influential seed sets from a new point of view:

$$\eta_{K,k} = \frac{\Gamma_{K,k} - K}{\tau_{K,k}}$$

Table 6 – Mean of τ values on K and k

$\bar{\tau}$	*GTaCB*	*TOPSIS*	*PR*	*DC*	*BC*	*CC*
Abrar	4.79	4.93	4.83	5.29	4.77	5.22
USAIR	5.91	4.03	4.03	3.99	4.16	4.04
EuroSiS	9.08	7.94	7.82	8.96	8.12	9.83
OCLinks	6.97	6.61	6.58	6.73	6.58	6.73
YEAST	10.12	9.70	9.68	9.85	9.72	10.18
Geom	11.19	10.95	11.03	11.45	10.61	11.93
HEP-th	15.26	14.19	14.30	14.75	14.62	15.98
PGP	15.08	14.74	14.35	15.62	14.64	16.29
Slashdot	11.04	10.52	10.45	10.74	10.51	10.72

Therefore, ***Table 7*** compares the average diffusion speeds each algorithm performs on the employed networks. It shows how the sets that GTaCB identifies, outperformed the sets of others; however, again in EuroSiS our algorithm didn't have better performance and this time, its value was only more than those of DC and CC, where PR peaked the diffusion speeds by far (approximately 44.5 nodes in each period).

Table 7 – Average η values of compared algorithms showing that how many nodes are averagely infected within a period.

$\bar{\eta}$	GTaCB	TOPSIS	PR	DC	BC	CC
Abrar	10.82	8.66	9.36	7.68	9.22	8.00
USAIR	5.81	3.34	3.53	3.29	4.17	3.29
EuroSiS	37.73	40.30	44.55	31.77	39.83	28.97
OCLinks	118.66	117.26	118.04	115.38	117.98	115.07
YEAST	85.68	82.80	83.76	80.87	83.84	77.31
Geom	137.93	124.84	125.54	115.81	132.42	108.65
HEP-th	88.87	82.55	85.77	77.08	82.26	69.32
PGP	155.83	138.61	142.39	125.45	146.59	118.95
Slashdot	1480.15	1340.20	1393.50	1240.94	1411.35	1212.33

6. Conclusion

Finding a set of influential nodes is of practical and theoretical importance in complex networks, specially to resolve a problem called "influence maximization". In this paper, we proposed GTaCB, a new algorithm to find the set of initial nodes by distributing the network into *K* sub-graphs via a community detection algorithm named, GCSC. And afterwards, by the implementation of a multi-attribute decision-making (MADM) technique known as, TOPSIS, to find the best node in each sub-graph with the aid of centrality measures as its attributes. The main novelty of the present paper is to cope with the influence maximization problem by utilizing four centrality measures of the nodes and considering the fact that the social networks are community based. To improve the previous studies in which the multi attribute techniques have been utilized, we used a community detection algorithm to separate the communities from each other to find the best influential nodes.

In order to evaluate the performance of GTaCB in comparison to other algorithms, we have simulated diffusion of the chosen seeds, identified by each approach, through the employment of an SIR model. The experimental results show that in one hand, the set of nodes that GTaCB suggests, has a higher diffusion quality in 8 out of 9 networks. On the other hand, despite its long diffusion process, in most of the cases it is faster than others in terms of diffusion speed, in particular when infection rate is sufficiently high ($k\alpha > 0.05$). Secondly, the results clearly illustrate that Jaccard Coefficient (*JC*) values between the sets identified by GTaCB and each of its rivals are considerably lower than *JC* values between the pair sets of the rest. It means that the majority of individual nodes, selected by the proposed algorithm, are not considered "influential" by the famous centrality measures. Apart from that, our experiments show that BC and PR compete rather effectively than DC and specially CC which was poorer in both diffusion quality and diffusion speed.

From an application point of view, it means that the detection of the network communities and the selection of the seed nodes (using the four well-known centrality measures) in each community is a nice strategy to find the influential nodes. Additionally, it is suggested to develop a mathematical programming model for the problem and to compare the results of the GTaCB with optimal solution on small-sized datasets to show the optimal gap.

References

1. Jackson, M.O., *Social and economic networks*. 2010: Princeton university press.
2. Jackson, M.O., *A Survey of Network Formation Models: Stability and Efficiency*, G. Demange and M. Wooders, Editors. 2003, Cambridge University Press: Cambridge. p. 11-57.
3. Agha Mohammad Ali Kermani, M., et al., *Introducing a procedure for developing a novel centrality measure (Sociability Centrality) for social networks using TOPSIS method and genetic algorithm.* Computers in Human Behavior, 2016. **56**: p. 295-305.
4. Alizadeh, M., C. Cioffi-Revilla, and A. Crooks, *Generating and analyzing spatial social networks.* Computational and Mathematical Organization Theory, 2017. **23**(3): p. 362-390.
5. Zhao, J., et al., *Competitive seeds-selection in complex networks.* Physica A: Statistical Mechanics and its Applications, 2017. **467**: p. 240-248.
6. Alizadeh, M. and C. Cioffi-Revilla. *Distributions of opinion and extremist radicalization: Insights from agent-based modeling*. in *International Conference on Social Informatics*. 2014. Springer.
7. Alizadeh, M., C. Cioffi-Revilla, and A. Crooks, *THE EFFECT OF IN-GROUP FAVORITISM ON THE COLLECTIVE BEHAVIOR OF INDIVIDUALS'OPINIONS.* Advances in Complex Systems, 2015. **18**(01n02): p. 1550002.
8. Alizadeh, M. and C. Cioffi-Revilla, *Activation regimes in opinion dynamics: Comparing asynchronous updating schemes.* Browser Download This Paper, 2015.
9. Alizadeh, M., et al., *Intergroup conflict escalation leads to more extremism.* 2012.
10. Guille, A., et al., *Information diffusion in online social networks.* ACM SIGMOD Record, 2013. **42**(1): p. 17-17.
11. Agha Mohammad Ali Kermani, M., A. Aliahmadi, and R. Hanneman, *Optimizing the choice of influential nodes for diffusion on a social network.* International Journal of Communication Systems, 2016. **29**(7): p. 1235-1250.
12. Chen, W., L.V.S. Lakshmanan, and C. Castillo, *Information and influence propagation in social networks.* Synthesis Lectures on Data Management, 2013. **5**(4): p. 1-177.
13. Chen, W., Y. Wang, and S. Yang. *Efficient influence maximization in social networks*. 2009. New York, New York, USA: ACM Press.
14. Kempe, D., J. Kleinberg, and É. Tardos. *Maximizing the spread of influence through a social network*. in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003. ACM.
15. Chen, W., C. Wang, and Y. Wang. *Scalable influence maximization for prevalent viral marketing in large-scale social networks*. 2010. New York, New York, USA: ACM Press.
16. Kermani, M.A.M.A., et al., *A novel game theoretic approach for modeling competitive information diffusion in social networks with heterogeneous nodes.* Physica A: Statistical Mechanics and its Applications, 2017. **466**: p. 570-582.
17. Carnes, T., et al. *Maximizing influence in a competitive social network*. New York, New York, USA: ACM Press.
18. Shang, J., et al., *CoFIM: A community-based framework for influence maximization on large-scale networks.* Knowledge-Based Systems, 2017. **117**: p. 88-100.
19. Borgatti, S.P., *Centrality and network flow.* Social networks, 2005. **27**(1): p. 55-71.
20. Borgatti, S.P. and M.G. Everett, *A graph-theoretic perspective on centrality.* Social networks, 2006. **28**(4): p. 466-484.
21. Opsahl, T., F. Agneessens, and J. Skvoretz, *Node centrality in weighted networks: Generalizing degree and shortest paths.* Social networks, 2010. **32**(3): p. 245-251.

22. Meng, F., et al. *Comparison of Different Centrality Measures to Find Influential Nodes in Complex Networks*. in *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*. 2017. Springer.
23. Du, Y., et al., *A new method of identifying influential nodes in complex networks based on TOPSIS.* Physica A: Statistical Mechanics and its Applications, 2014. **399**: p. 57-69.
24. Newman, M.E., *Modularity and community structure in networks.* Proceedings of the national academy of sciences, 2006. **103**(23): p. 8577-8582.
25. Chen, Y.-C., et al., *CIM: Community-based influence maximization in social networks.* ACM Transactions on Intelligent Systems and Technology (TIST), 2014. **5**(2): p. 25.
26. Domingos, P. and M. Richardson. *Mining the network value of customers*. in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. 2001. ACM.
27. Newman, M.E.J., *Spread of epidemic disease on networks.* Physical Review E, 2002. **66**(1): p. 16128-16128.
28. Newman, M.E.J. and J. Park, *Why social networks are different from other types of networks.* Physical Review E, 2003. **68**(3): p. 36122-36122.
29. Hethcote, H.W., *Qualitative analyses of communicable disease models.* Mathematical Biosciences, 1976. **28**(3-4): p. 335-356.
30. Leskovec, J., et al. *Cost-effective outbreak detection in networks*. 2007. New York, New York, USA: ACM Press.
31. Goyal, A., W. Lu, and L.V.S. Lakshmanan. *CELF++:optimizing the greedy algorithm for influence maximization in social networks*. 2011. New York, New York, USA: ACM Press.
32. Liu, B., et al., *Influence Spreading Path and Its Application to the Time Constrained Social Influence Maximization Problem and Beyond.* IEEE Transactions on Knowledge and Data Engineering, 2014. **26**(8): p. 1904-1917.
33. Yang, W.-S. and S.-X. Weng, *Application of the Ant Colony Optimization Algorithm to the Influence-Maximization Problem.* International Journal of Swarm Intelligence and Evolutionary Computation, 2012. **1**: p. 1-8.
34. Bucur, D. and G. Iacca. *Influence maximization in social networks with genetic algorithms*. in *European Conference on the Applications of Evolutionary Computation*. 2016. Springer.
35. Jiang, Q., et al. *Simulated Annealing Based Influence Maximization in Social Networks*. in *AAAI*. 2011.
36. Gong, M., et al., *Influence maximization in social networks based on discrete particle swarm optimization.* Information Sciences, 2016. **367-368**: p. 600-614.
37. Gandomi, A.H., X.-S. Yang, and A.H. Alavi, *Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems.* Engineering with Computers, 2013. **29**(1): p. 17-35.
38. Mesgari, I., et al., *Identifying Key Nodes in Social Networks Using Multi-Criteria Decision-Making Tools*, in *Mathematical Technology of Networks*. 2015, Springer. p. 137-150.
39. Fox, W.P. and S.F. Everton, *Using Mathematical Models in Decision Making Methodologies to Find Key Nodes in the Noordin Dark Network.* American Journal of Operations Research, 2014. **04**(04): p. 255-267.
40. Zhang, W., Q. Zhang, and H. Karimi, *Seeking the Important Nodes of Complex Networks in Product R&D Team Based on Fuzzy AHP and TOPSIS.* Mathematical Problems in Engineering, 2013. **2013**: p. 1-9.

41. Jaquet, M. and V. Pechal, *Lecture with Computer Exercises : Modelling and Simulating Social Systems with MATLAB Epidemic spreading in a social network.* Lecture, 2009(December).
42. Cao, T., et al. *OASNET: an optimal allocation approach to influence maximization in modular social networks*. in *Proceedings of the 2010 ACM Symposium on Applied Computing*. 2010. ACM.
43. Zhang, X., et al., *Identifying influential nodes in complex networks with community structure.* Knowledge-Based Systems, 2013. **42**: p. 74-84.
44. Lancichinetti, A., S. Fortunato, and F. Radicchi, *Benchmark graphs for testing community detection algorithms.* Physical Review E, 2008. **78**(4): p. 46110-46110.
45. Wang, Y., et al. *Community-based greedy algorithm for mining top-K influential nodes in mobile social networks*. New York, New York, USA: ACM Press.
46. Lancichinetti, A., S. Fortunato, and F. Radicchi, *Benchmark graphs for testing community detection algorithms.* Physical review E, 2008. **78**(4): p. 046110.
47. Orman, G.K., V. Labatut, and H. Cherifi, *On Accuracy of Community Structure Discovery Algorithms.* Journal of Convergence Information Technology, 2011. **6**(11): p. 283-292.
48. Karimi-Majd, A.-M., M. Fathian, and B. Amiri, *A hybrid artificial immune network for detecting communities in complex networks.* Computing, 2015. **97**(5): p. 483-507.
49. Hespanha, J.P., *An efficient matlab algorithm for graph partitioning.* Santa Barbara, CA, USA: University of California, 2004.
50. Hwang, C.-L. and K. Yoon, *Methods for Multiple Attribute Decision Making*. 1981. p. 58-191.
51. Kermani, M.A.M.A., H. Navidi, and F. Alborzi, *A novel method for supplier selection by two competitors, including multiple criteria.* International Journal of Computer Integrated Manufacturing, 2012. **25**(6): p. 527-535.
52. Ghaseminejad, A., H. Navidi, and M. Bashiri, *Using Data Envelopment Analysis and TOPSIS method for solving flexible bay structure layout.* International Journal of Management Science and Engineering Management, 2011. **6**(1): p. 49-57.
53. Ahmadi, M., S.M.H. Molana, and S.M. Sajadi, *A hybrid FMEA-TOPSIS method for risk management, case study: Esfahan Mobarakeh Steel Company.* International Journal of Process Management and Benchmarking, 2017. **7**(3): p. 397-408.
54. Mahmoodzadeh, S., et al., *Project selection by using fuzzy AHP and TOPSIS technique.* International Journal of Human and social sciences, 2007. **1**(3): p. 135-140.
55. Wei, J., *TOPSIS Method for Multiple Attribute Decision Making with Incomplete Weight Information in Linguistic Setting.* Journal of Convergence Information Technology, 2010. **5**(10): p. 181-187.
56. Hu, J., et al., *A modified weighted TOPSIS to identify influential nodes in complex networks.* Physica A: Statistical Mechanics and its Applications, 2016. **444**: p. 73-85.
57. Agha Mohammad Ali Kermani, M., et al., *A note on predicting how people interact in attributed social networks*. 2014. p. 2510-2514.
58. Boguñá, M., et al., *Models of social networks based on social distance attachment.* Physical Review E, 2004. **70**(5): p. 56122-56122.
59. Jaccard, P., *THE DISTRIBUTION OF THE FLORA IN THE ALPINE ZONE.1*. New Phytologist, 1912. **11**(2): p. 37-50.

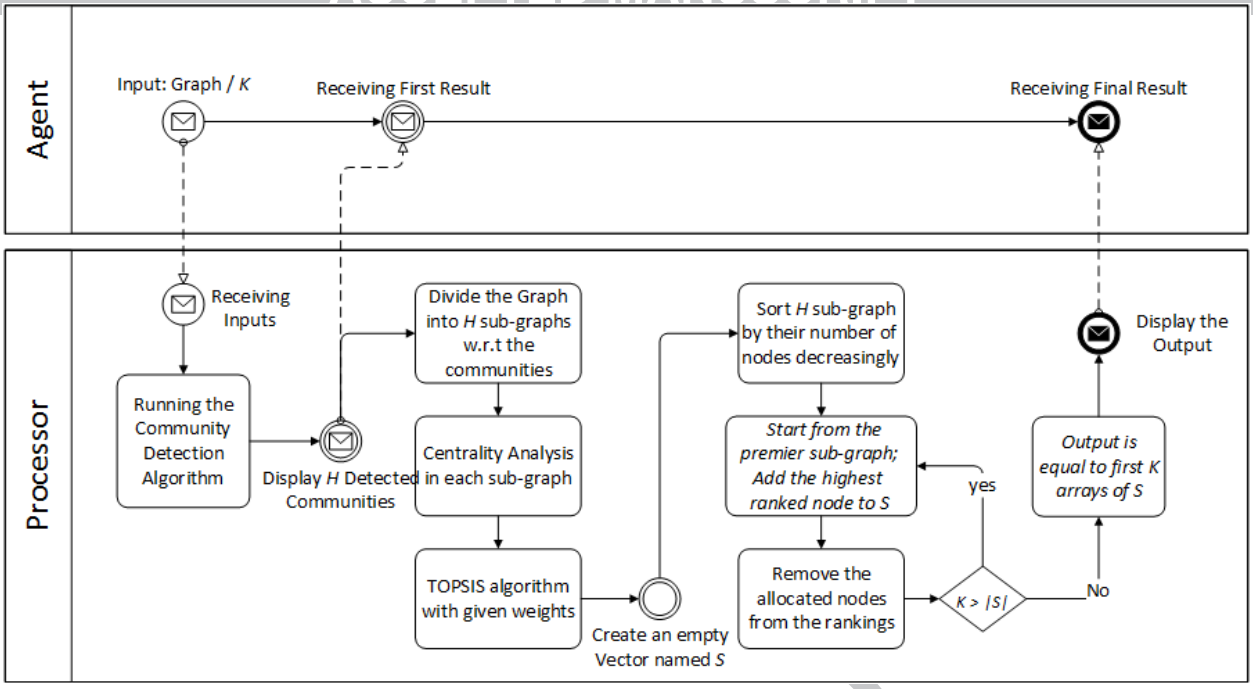


Fig.1 – The schematic process of GTaCB algorithm

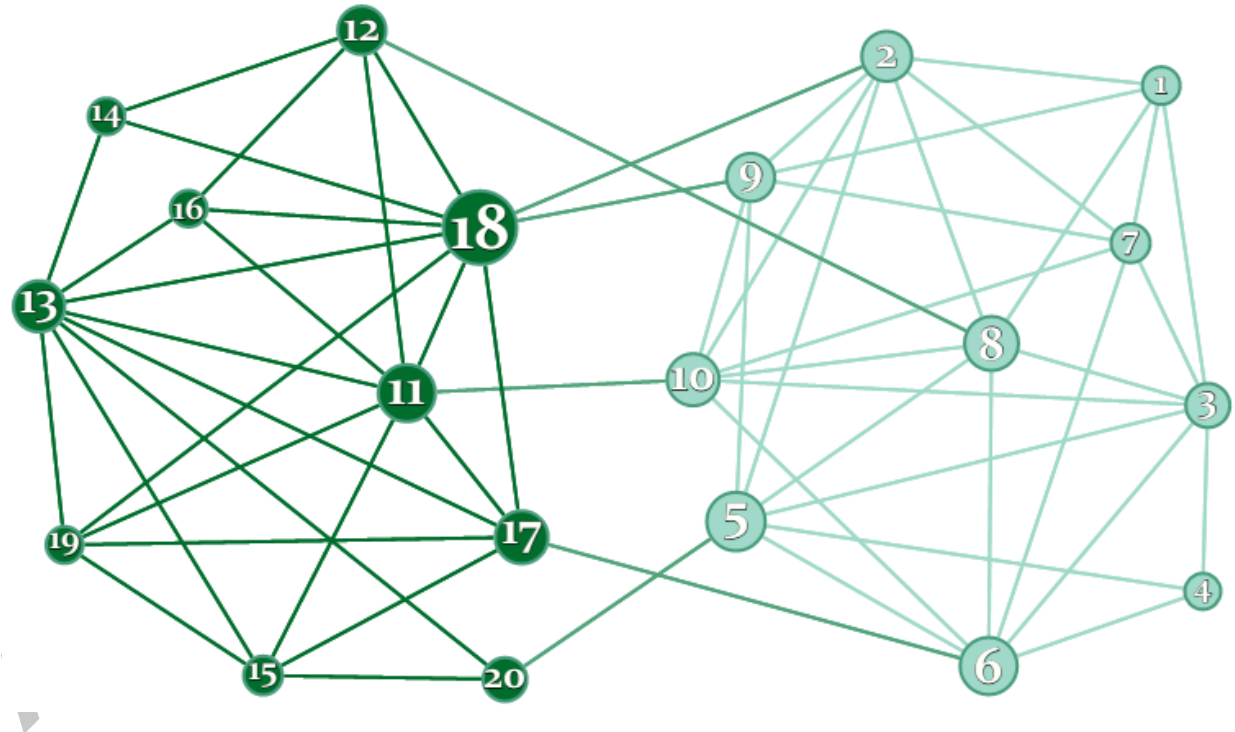


Fig.2 – Ex1 with two communities

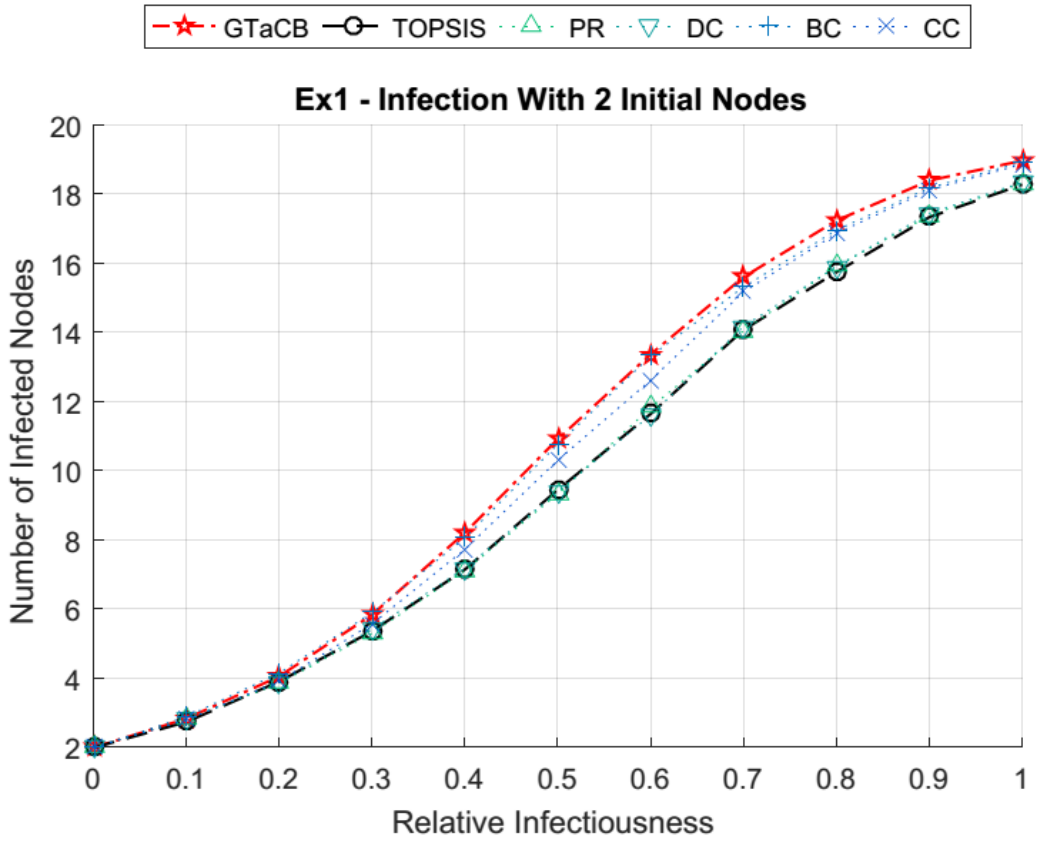


Fig.3 – Ex1 SIR infection results with 2 seeds

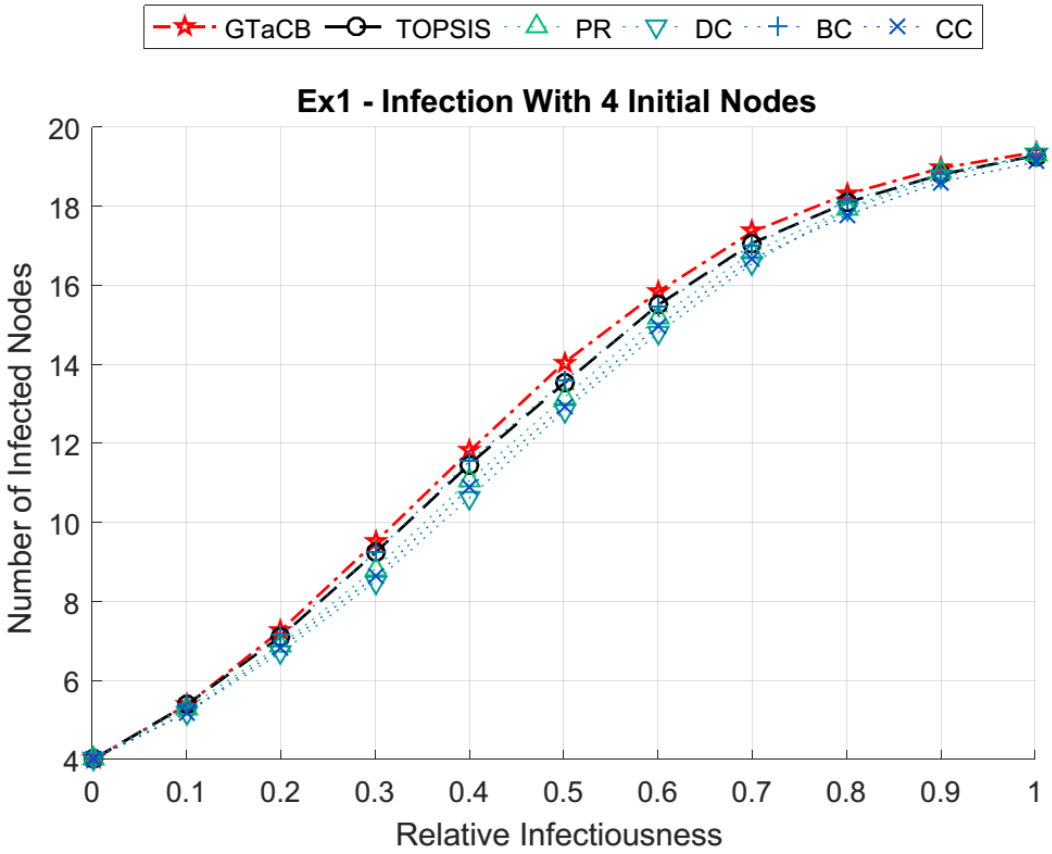


Fig.4 - Ex1 SIR infection results with 4 seeds

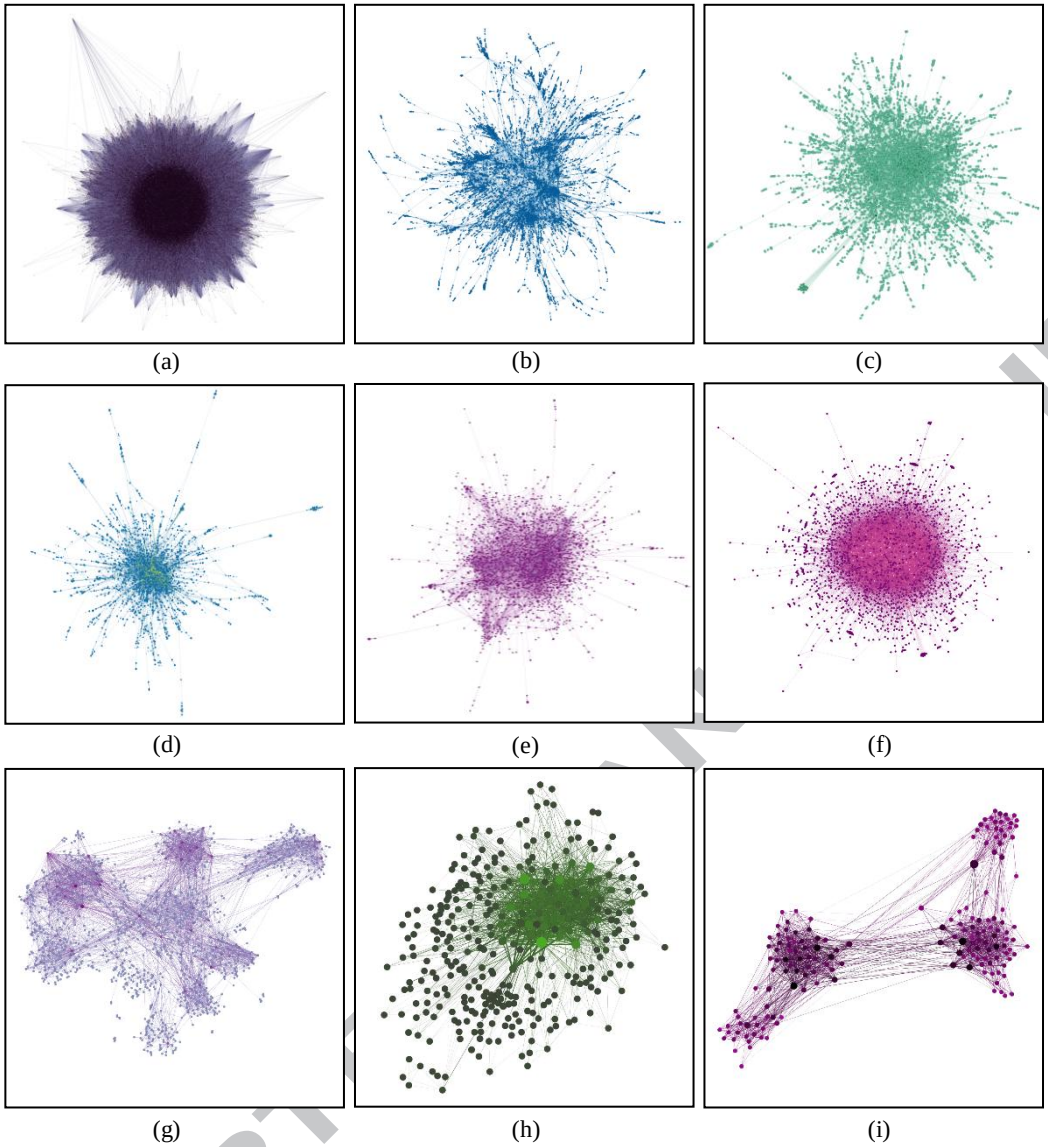


Fig. 5 - The visualizations of Slashdot(a), PGP (b), HEP-th(c), Geom(d), Yeast(e), OCLinks(f), EuroSiS(g), USAir(h) and Abrar(i) graphs – The layouts are based on Force-Atlas algorithm

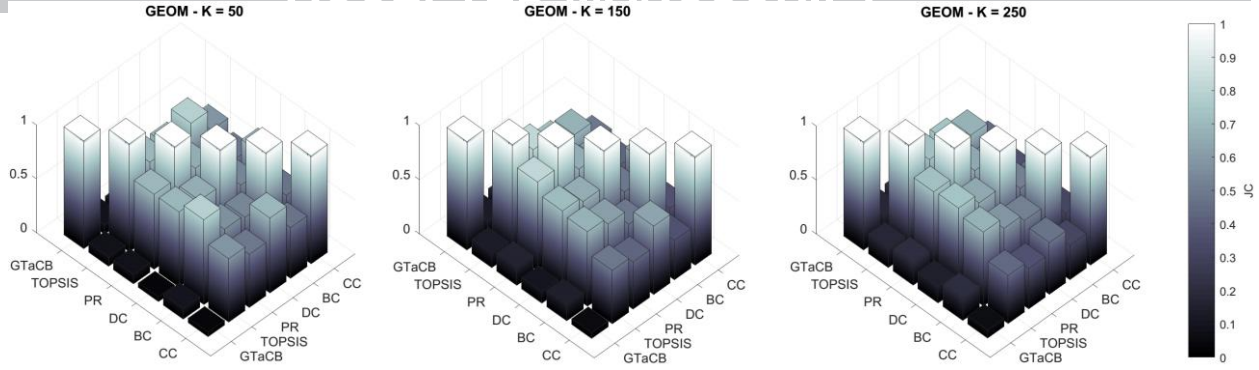


Fig. 6 – PGP graph with 50 (right) and 250 (left) detected communities. The colors represent different identified communities

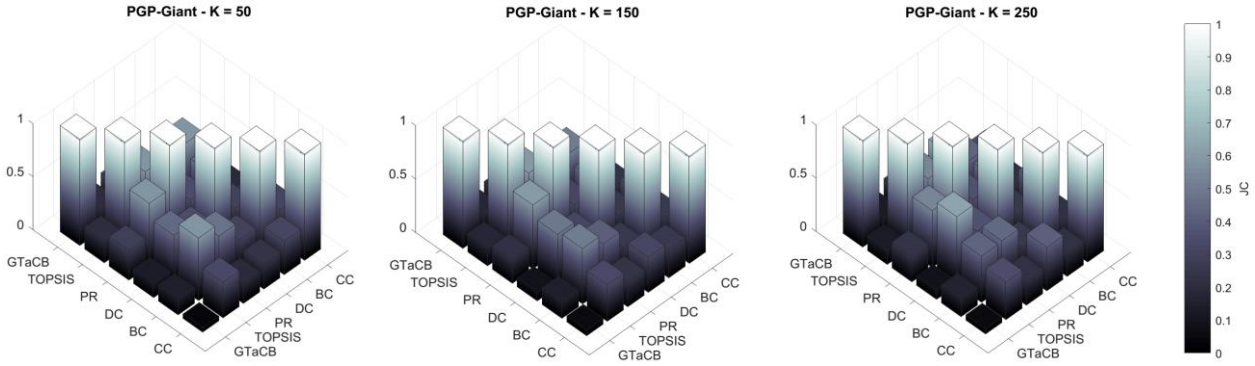
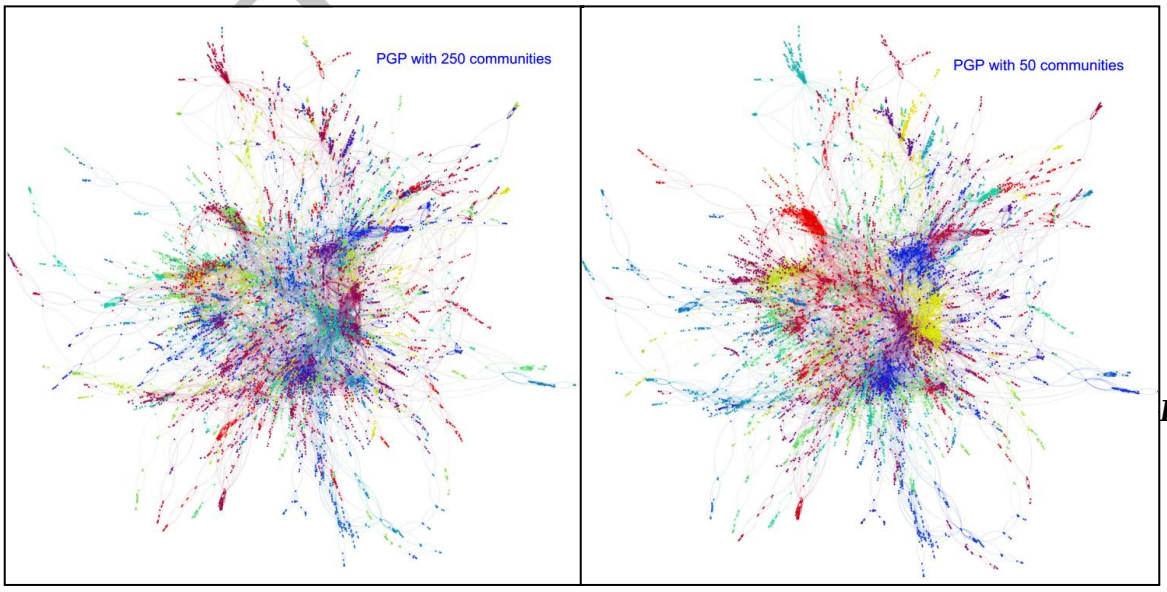


Fig. 7 – The relationship between each pair of utilized algorithms' detected seeds in PGP network



EOM

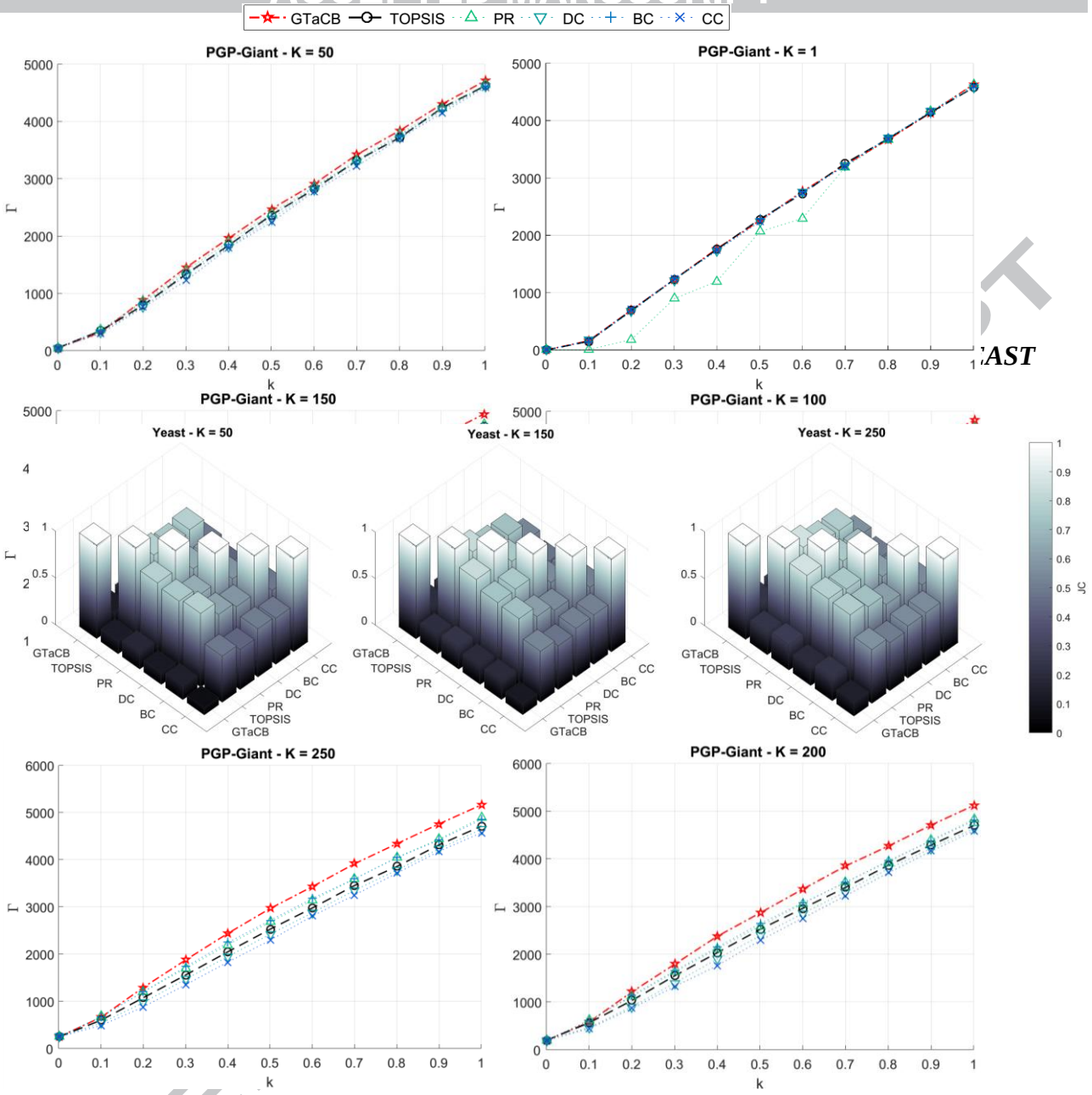


Fig. 10 – Infection result comparisons in PGP Network

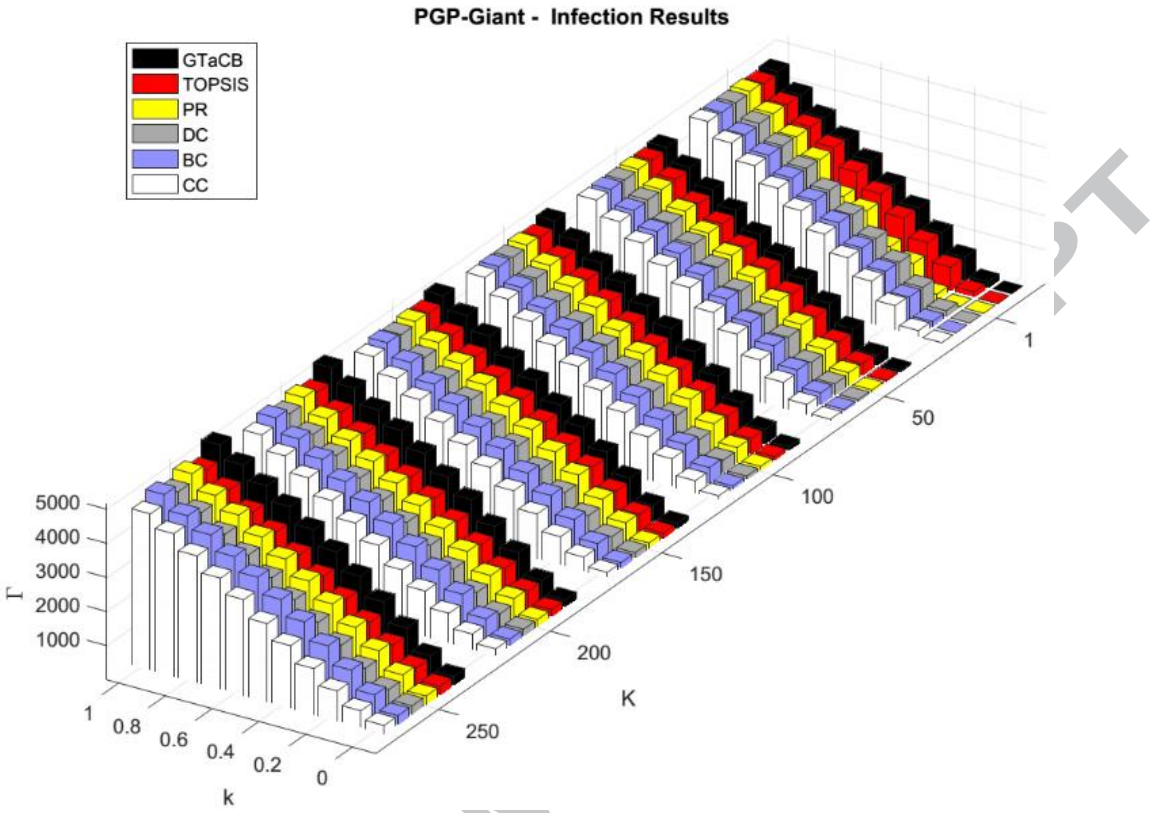


Fig. 11 – Infection result comparisons in 3D-bar chart of PGP Network

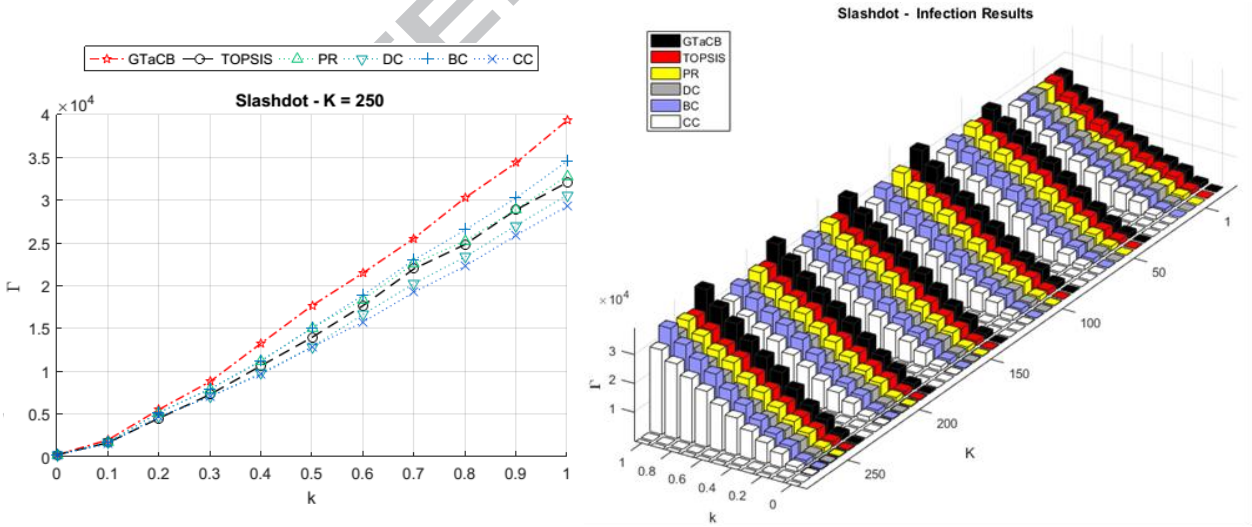


Fig. 12 – Infection result comparisons in Slashdot Network

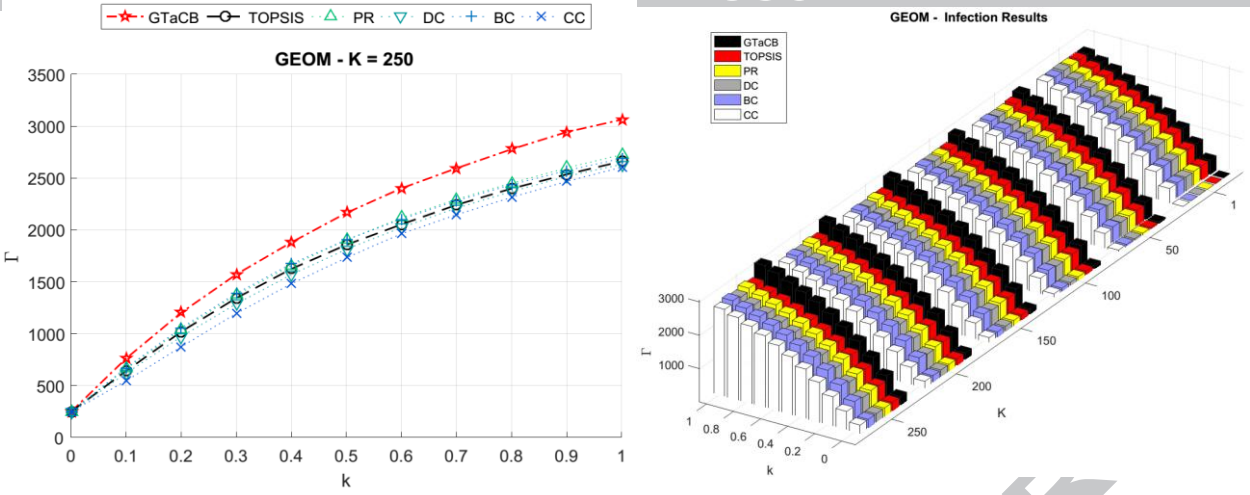


Fig. 13 – Infection result comparisons in HEP-th Network

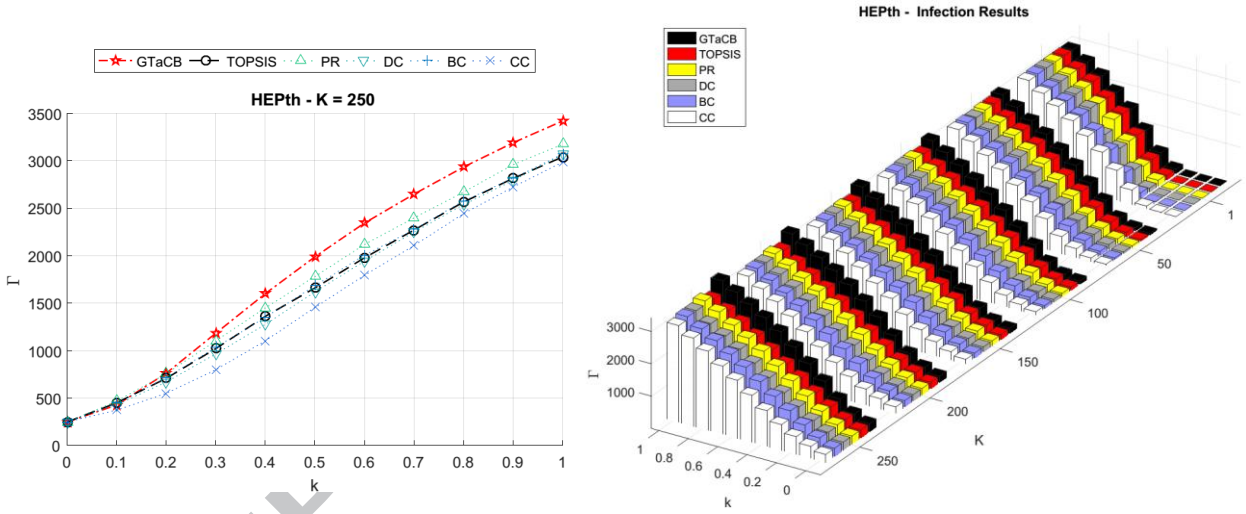


Fig. 14 – Infection result comparisons in GEOM Network

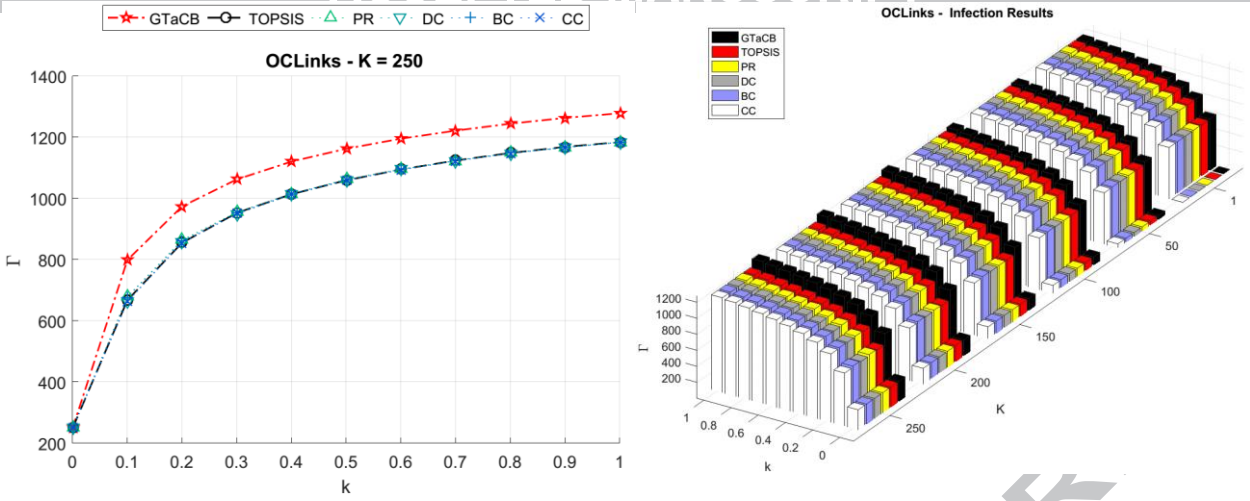


Fig. 15 – Infection result comparisons in YEAST network

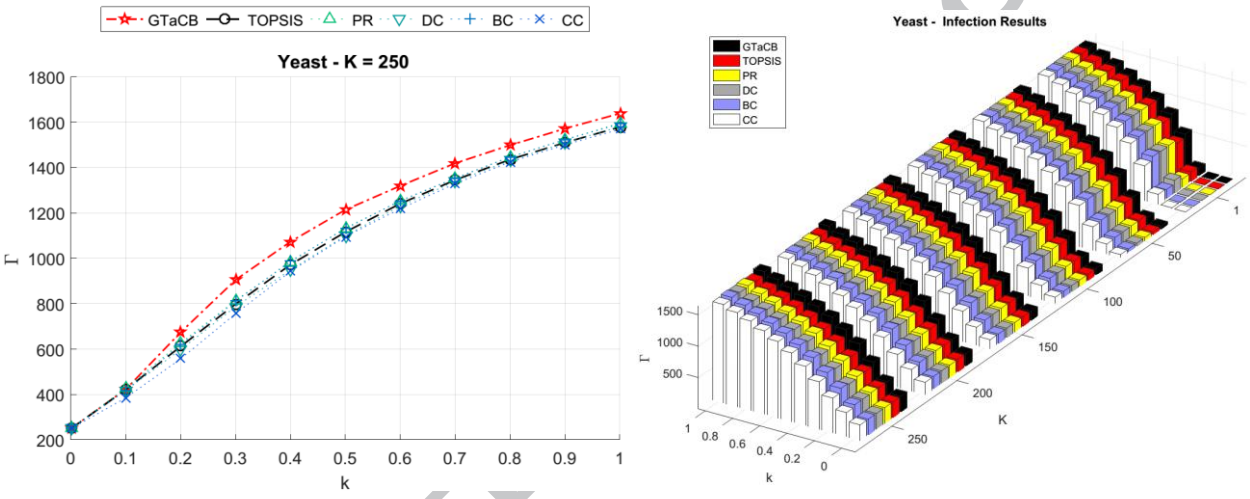


Fig. 16 – Infection result comparisons in OCLinks Network

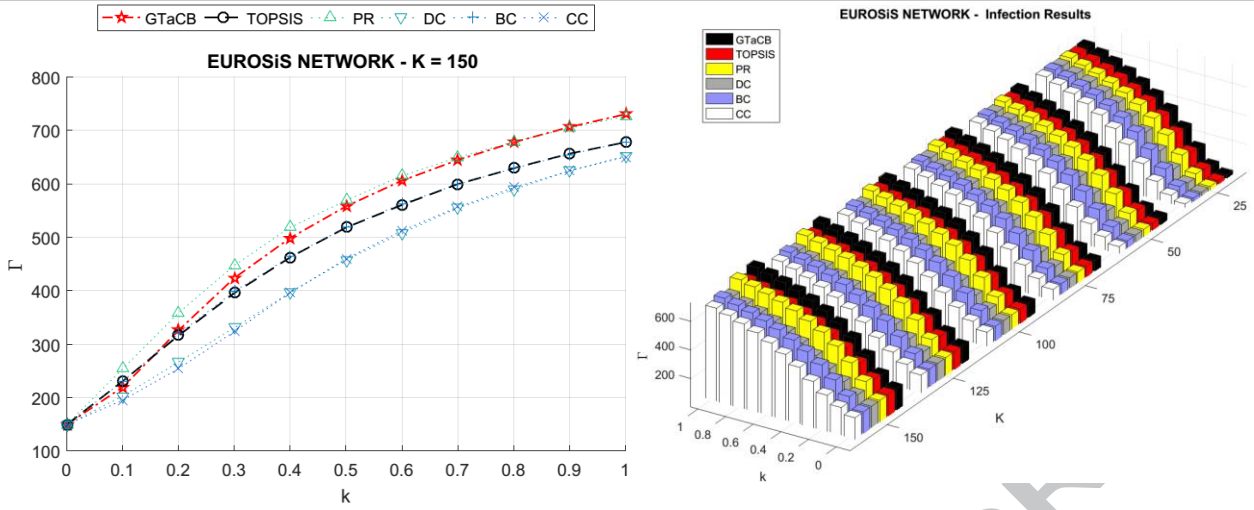


Fig. 17 – Infection result comparisons in USAir Network

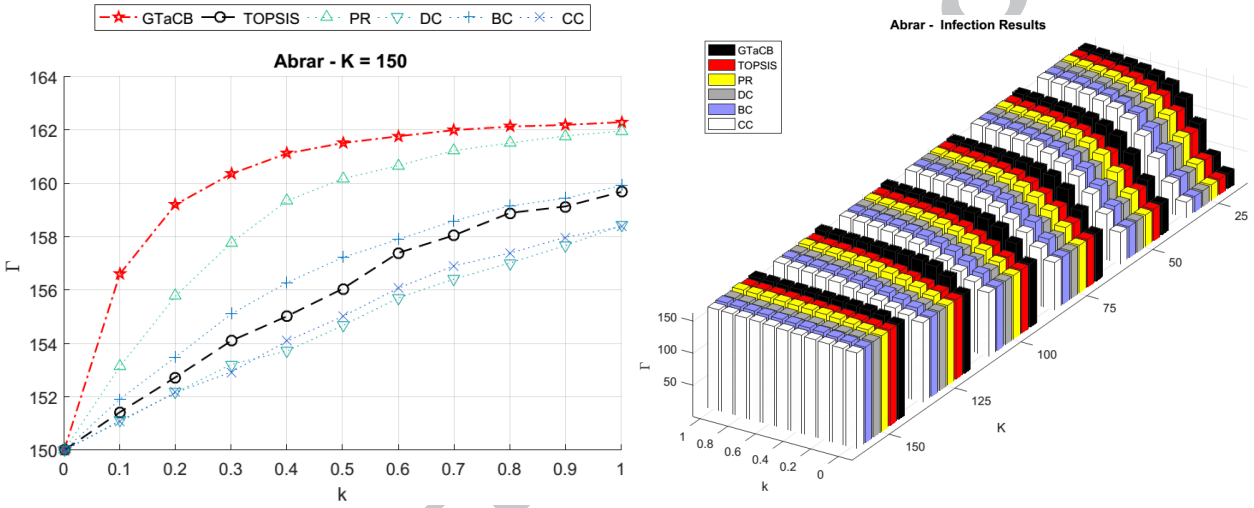


Fig. 18 – Infection result comparisons in Abrar Network

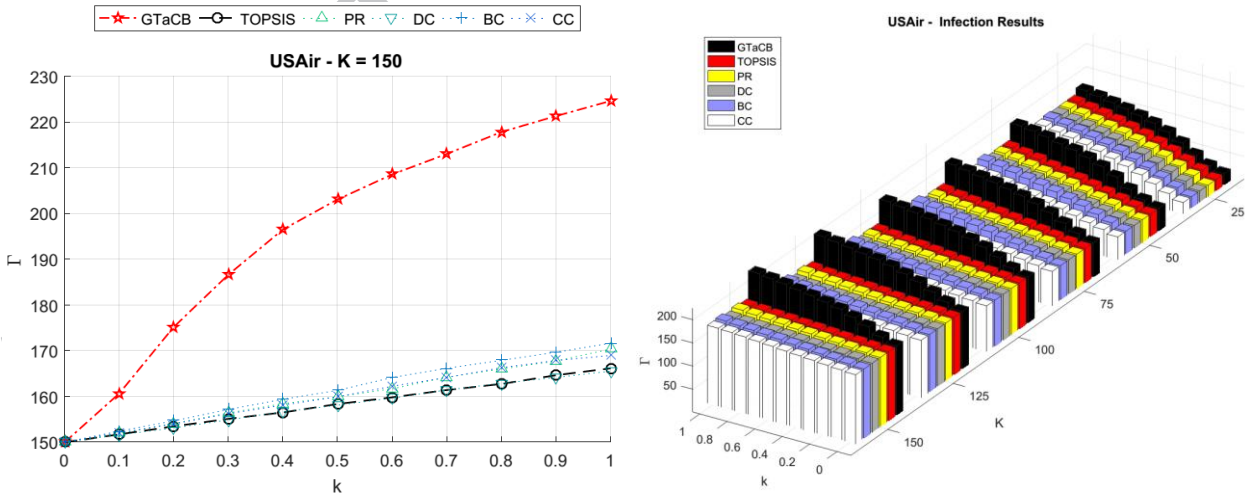


Fig. 19 – Infection result comparisons in EuroSiS Network

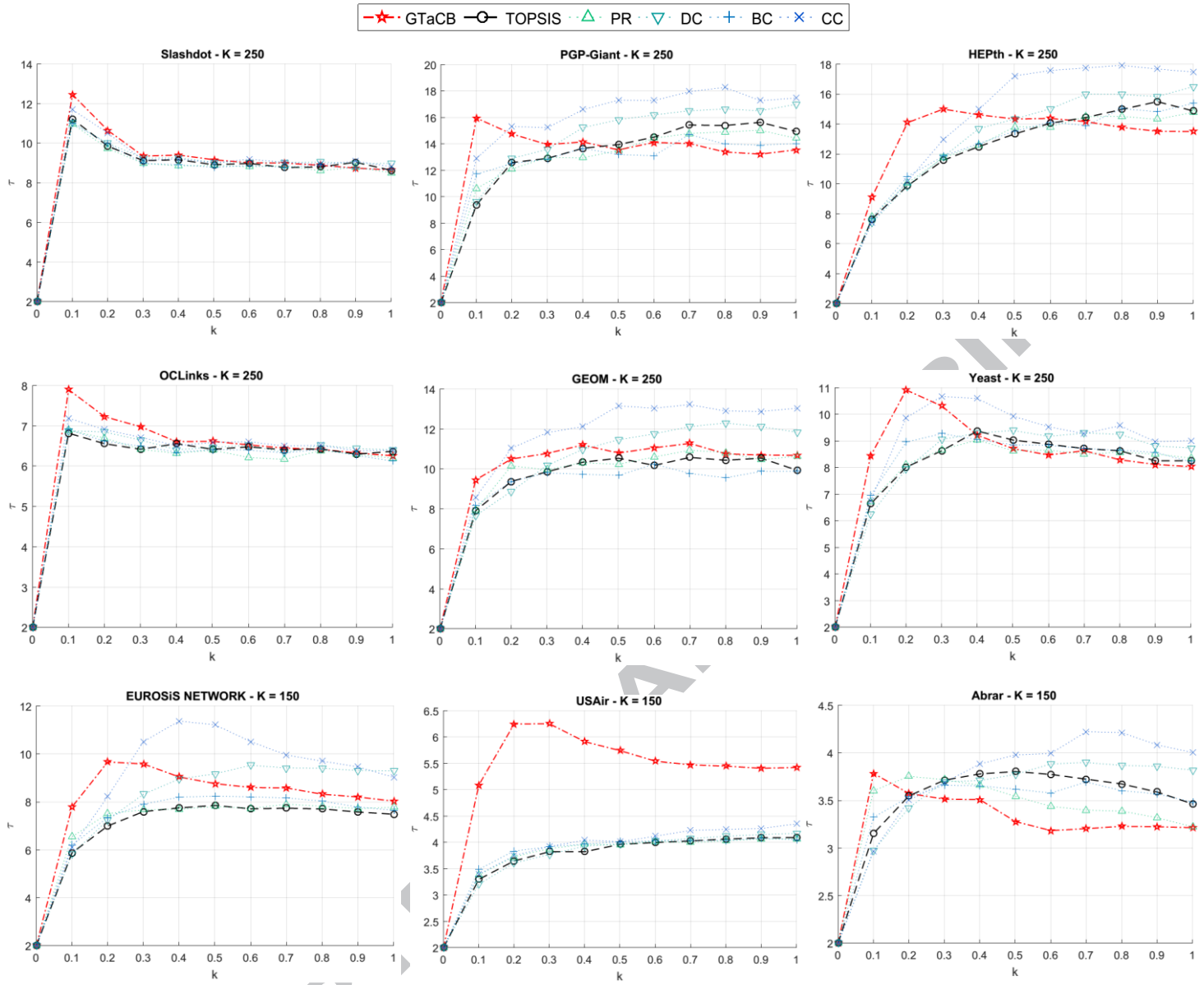


Fig. 20 –Comparison of τ values for the employed networks

Highlights

- A novel greedy and hybrid algorithm proposed for influence maximization.
- The proposed algorithm is based on Hespanha and Topsis.
- The efficiency of the algorithm is evaluated using eight datasets.