

# Exploring Intelligent Approach of Influence Minimization Considering the Node Surveillance in Online Social Networks

JO CHERIYAN (✉ [jocheriyan@am.amrita.edu](mailto:jocheriyan@am.amrita.edu))

Amrita Vishwa Vidyapeetham - Amritapuri Campus <https://orcid.org/0000-0002-4154-3963>

Sajeev G P

Amrita University: Amrita Vishwa Vidyapeetham

---

## Research

**Keywords:** influence minimization, social networks, negative information, live nodes

**Posted Date:** March 10th, 2021

**DOI:** <https://doi.org/10.21203/rs.3.rs-101760/v2>

**License:**   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

# Exploring Intelligent Approach of Influence Minimization considering the Node Surveillance in Online Social Networks

Jo Cheriyan

Received: date / Accepted: date

**Abstract** Attractive information such as innovations, awareness campaigns, branding, and advertising help people positively. Whereas, awful information such as rumors, malicious viruses, pornography, and revenge disturb people. The negative information contributes to chaos among people; therefore, it is to be blocked and hinder from further diffusion. This has motivated us towards the study of the problem named *influence minimization*. As the real world network can be modeled to a multilayer network, we focus our study towards the information diffusion through a multilayer network. Each node assigns a threshold, and its variation affects the rate of influence propagation across the network. In the *influence minimization* problem, the energy level of each node changes that help to formulate the function that minimizes the influence propagation. By applying two reduction policies, we are able to optimize our objective of minimizing the influence towards repulsive information. In this article, we consider the user response and its surveillance in the network. Repeated experiments on real networks has helped us to validate the proposed methods.

**Keywords** influence minimization · social networks · negative information · live nodes.

## 1 Introduction

A complex network [1] is an explicit model for a real-world system such as technological networks, social networks, business networks, and biological networks

---

Jo Cheriyan  
Department of Computer Science, School of Engineering  
Amrita Vishwa Vidyapeetham  
Tel.: +91-476-280-2100  
Fax: +91-944-604-8103  
E-mail: jocheriyan@am.amrita.edu

[24]. Online social networks create gratifying support for sharing information among people globally. Some examples of social networks are Facebook, Twitter, Instagram, google+ and LinkedIn. Social Networks act as a medium for propagating information across the world. The information includes both the positive information such as novel ideas, interesting news, and advertising, and the negative information such as disinformation, hateful rumors, and malicious virus, etc. The positive information propagates along the length and breadth of the networks and impacts most of the entities, which is named as *influence maximization* [7], [4]. It is commonly observed that the negative information gets minimal care to spread widely. This paves to a design strategy for diminishing the influence advancement, called *influence minimization* [31], [33].

The Influence Minimization intent to block or remove nodes, which are capable of spreading negative information in each of the social community. The impact can be restricted by deleting edges to block malicious nodes. Minimizing the propagation of awful information in a networked graph is a challenging issue. Influence minimization is accomplished by analyzing the entropy of each node capable of inflating the spread. In general, the influence minimization performs by cutting the edges that lead to dis-joined the nodes, no matter where the nodes are or how much capacity to carry over the information. Thus we can attain our primary objective of protecting society from untruthful facts and troubles.

The influence minimization finds its existence in many applications across various domains like social media, epidemiology, security, and public health. Here we coin a novel concept for the problem and propose a term *entropy based Influence Minimization (e-IM)*. The seed nodes(k) [8] starts propagating information which follows the process of *Influence Maximization* [7]. The *e-IM* model select  $m$  nodes which tend to be propagated negative information. These sets of nodes that propagate awful information are called malicious nodes. The nodes between the seed set and the malicious set are removed from the network thereby the subgraphs that link to malicious nodes are disjoined temporarily. The immunization is an efficient process for controlling diseases by isolating the infected one. Restrict the spreading of rumors in a limited area, and prevent the infection by a computer virus.

The surveillance nodes are the optimal subset of nodes that advertise the spreading information. When the transfusion of rumorss or viruses is being reported, it helps to shape the social networks based on an epidemic outbreak. The function of an energy level of the infected node forms the subset of the surveillance nodes. Hence, it affects the estimation of the source and outbreak path. This helps in early detection of the spreading of awful information. Therefore designing a minimization algorithm contributes to maximizing the accuracy of surveillance function, that leads to the social well-being by preventing unwanted information.

In this article, we propose the study of influence minimization problem by node surveillance. It is carried out by the function of the source of information. When a node receives information, its threshold energy increases [6]. Energy

spikes are measure in terms of entropy. The influence maximization algorithm generates a seed set ( $k$ ) for maximizing the information explicitly. The  $e$ - $IM$  algorithm maximizes the threshold function and forms a subset of nodes, called malicious nodes( $m$ ). Applying Targen’s algorithms, we identify the articulation nodes. Let  $p$  be the number of nodes, we design a  $k$ - $m$  policy, remove all common nodes from the graph to make the two sets are disjoint. We further verify that the two sets  $p$  and  $k$ - $m$  are disjoint. This helps in performance improvement of our method.

The  $k$ - $m$  policy and  $((k-m)-p)$  policy improve our algorithm to reduce the influence to a great extent. In both cases, the principle of the disjoint subset helps to improve the  $e$ - $IM$  problem and help to attain linear running time by considering network size. To show the reliability and effectiveness of the  $e$ - $IM$  algorithm, we conduct experiments with the help of a python-based simulator on real-world networks. We could establish that our proposed algorithm and policies are better than any of the state-of-the-art. It is faster in magnitude than any of the similar kind. To summarize the article, the major contributions are:

- Propose a novel *influence minimization* algorithm called  $e$ - $IM$ .
- Formulate two policies ( $k$ - $m$ ) and  $((k-m)-p)$  for establishing  $e$ - $IM$  algorithm.
- Validating the algorithm with real-network for showing the effectiveness of  $e$ - $IM$ .

The rest of this paper is organized as follows. Section 2 familiarize some related work in the field of influence minimization, it gives an insight into the literature. Section 3 describes the background concepts in the proposed methods that help to understand the concept thoroughly. The problem and proposed procedure to downsize the influence of information diffusion in social networks, and it described in section 4 and its validation method is done in section 5. Section 6 illustrate the results. Finally, the article is concluded in section 7.

## 2 Related Work

There are two broad types of information propagation across social networks. One category is positive information that is beneficial to the community where the other is malicious information. The first category has to propagate wildly, whereas the other one has to minimize. The influence of maximization and influence minimization are the two research problems in social networks.

The influence minimization problem reduces the propagation of rumors or disinformation by blocking nodes from a topic modelling perspective [34]. When undesirable events propagate in a social network, reduce the size of the infected volume by blocking some nodes outside the infection area. This optimization problem makes use of HDA-LDA and KL to analyze the influence in topic modeling in the independent cascade model [2]. The topic-aware influence minimization approach works based on betweenness centrality and the

concept of out-degree. We observed that this approach is better than any of the centrality based approach but especially at the beginning of the contamination.

The *targeted influence minimization* is intended to minimize the influence of negative information to some particular class of user groups in social networks [31]. The algorithm focus on two cases of influence minimization problems, the first one is the impact of budget, and the second one is robust sampling [21], [17]. The algorithm provides an optimal solution and greedy approximation. Both are not appropriate for large dynamic networks such as online social networks. The robust sampling method applies to real social networks that guaranteed an effective solution. The sampling-based solution covers the maximum area where the information spreads. But the method is less efficient when the incremental addition of nodes having awful information.

Another approach is the minimization of rumors spreading by considering user experience dynamically [30]. The method is to plug off a subset of the nodes so that malicious rumors can be blocked. The constraint of the user experiences is also taken care of by the minimization problem. A threshold is applied to each node and is called the time-stamp, the blocking time. If a user exceeds the threshold, the service of the network system declined. Based on the above restrictions, a problem has defined and proposed a solution based on survival theory, and maximum likelihood principle [26]. The popularity and user inclination towards rumors analyze through Ising Model [12]. The algorithm became lesser accurate when the structure of the networks become more complex. Each node examines closely, and its surveillance can be incorporated to enhance the effectiveness of influence minimization.

The influence minimization problem is studied in the Linear Threshold(LT) model [32], [33]. The model consists of directed graphs where each node can be of two states: active or inactive. Information propagation progresses, the status of the node can change from inactive to active. Two scenarios for the influence minimization has analyzed. The first one is to cut the supply of products into distributors to minimize the sale of faulty products. The second is to block the information at the node level for minimizing further propagation. The integer programming problem helps for the above cases for an optimal solution. The optimal solution generated failed to scale for the large graph or multilayer graphs.

The study on the *Influence Minimization* in online social networking is challenging. When we consider real-world networks, layers arrange in the form of a stack for building a model. The information diffusion across the layers has to minimize. We observe that none of the present methods are capable of minimizing the influence spread in dynamic complex networks. This has been motivated us to design an effective method that blocks widespread rumors and awful information.

### 3 Preliminaries

In this section, we discuss some basic concepts of mathematical representation of complex networks, diffusion model, and influence minimization. Here we consider multilayer networks. All these would help the reader to understand the proposed work.

#### 3.1 Multilayer Networks

A monolayer network represent as a graph with tuples,  $G = (V, E)$ , where  $V$  is the set of vertices,  $E$  is the set of edges [23]. This mathematical representation can easily be extended to a network with multiple layers. A multilayer graph is expressed as a quadruple,

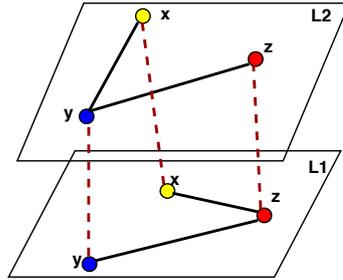
$$M = (V_M, E_M, V, L) \quad (1)$$

where  $V$  is the set of  $N$  vertices, and  $L$  is the set of elementary layers [27]. The layer  $L_m^d$  in which  $m$  is the total number of layers and  $d$  is the *aspect* of a network [20]. The notation  $L = \{L_m^d\}_{m=1}$  stands for stack of layers in the networks described through  $d$  aspect. We could build up layers in a multilayer network by integrating a set of all possible combinations of elementary layers  $L_1 \times \dots \times L_d$ . The notation  $V_M$  denote the set of vertices in each layer in the network,  $V_M \subseteq V \times L_1 \times \dots \times L_d$ , and  $E_M$  is the interconnection edges,  $E_M \subseteq V_M \times V_M$ . The set of vertices of layer  $\alpha$  is denoted as

$$V_\alpha = \{v_1^\alpha, v_2^\alpha, \dots, v_n^\alpha\} \quad (2)$$

The interconnection edge between layers  $\alpha$  and  $\beta$  is defined as,

$$E_M = \{E_{\alpha\beta} \subseteq V_\alpha \times V_\beta; \alpha, \beta = \{1, 2, \dots, m\}, \alpha \neq \beta\} \quad (3)$$



**Fig. 1** A multiplex network with  $V = \{x, y, z\}$  and  $L = \{L_1, L_2\}$

### Supra-Adjacency Matrix

The adjacency matrix for each layer  $G_\alpha$  is represented as  $A_{dj}^\alpha = a_{ij}^\alpha$ , where

$$a_{ij}^\alpha = \begin{cases} 1, & \text{if } \langle v_i^\alpha, v_j^\alpha \rangle \in E_\alpha \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where  $1 \leq i, j \leq N_\alpha$  and  $1 \leq \alpha \leq m$ . The cross layer or inter layer adjacency matrix corresponding to  $E_{\alpha\beta}$  is the matrix  $A_{dj}^{[\alpha\beta]} = (a_{ij}^{\alpha\beta})$  is given by

$$a_{ij}^{\alpha\beta} = \begin{cases} 1, & \text{if } \langle v_i^\alpha, v_j^\beta \rangle \in E_{\alpha\beta} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

The matrix representation for each layer is  $G^{[\alpha]}$ , and the supra adjacency matrix  $A^{[\alpha]}$  is  $G^{[\alpha]}(V, E)$  [5]. Where each element

$$\bar{a}_{ij} = \begin{cases} 1, & \text{if } a_{ij}^\alpha = 1 \text{ for some } 1 \leq \alpha \leq m \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

It is a symmetric matrix with order  $N^{[\alpha]}$ , number of nodes in the layer  $\alpha$ . Where each element,  $a_{ij}^{[\alpha]} = 1$  if and only if there is an edge between  $i$  and  $j$  in  $G^{[\alpha]}$ . And the coupling matrix,  $C \leftarrow G_c = \{C_{ij}\}$  an  $N \times N$  matrix with  $C_{ij} = 1$  if and only if the same node in different layers [10].

**Table 1** Supra-Adjacency matrix for the multiplex network shown in Fig. 1

$$A_{Ms} = \begin{pmatrix} & x \otimes l_1 & y \otimes l_1 & z \otimes l_1 & x \otimes l_2 & y \otimes l_2 & z \otimes l_2 \\ x \otimes l_1 & 0 & 0 & 1 & 1 & 0 & 0 \\ y \otimes l_1 & 0 & 0 & 1 & 0 & 1 & 0 \\ z \otimes l_1 & 1 & 0 & 0 & 0 & 0 & 1 \\ x \otimes l_2 & 1 & 0 & 0 & 0 & 1 & 0 \\ z \otimes l_2 & 0 & 1 & 0 & 1 & 0 & 1 \\ y \otimes l_2 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

The Supra-Laplacian matrix could be derived from supra-adjacency matrix. The Supra-Laplacian matrix represents the multilayer graphs which disclose major spectral properties such as coupling strength between layers and dynamic diffusion process [13]. The Supra Laplacian matrix  $\bar{L} = \bar{D} - \bar{A}$  where  $\bar{D}$  is the diagonal matrix and  $\bar{A}$  is the adjacency matrix [11].

### 3.2 Linear Threshold Model

A Linear Threshold(LT) model is represented as 4-tuples,  $(V_N, E_N, \theta, W)$  where  $V_N$  is the set of  $n$  nodes,  $E_N = V_N \times V_N$  is directed edges [19]. Additional parameter  $\theta$  is the threshold value,  $\theta_i = (0, 1]$  and The function  $W$  assign an influence weight,  $W(i, j) \in (0, 1]$  to each  $edge(i, j) \in E$ . As this model scale

to multilayer networks, number of tuple added to represents multilayer networks. The function assign weight to each edges either in a layer or between the layers. The *weighted supra-adjacency matrix* is to be formed for the simulation purpose. The in-neighbor set and out-neighbor set of each node are to be defined and  $Edge(i,j)$  represents that node  $j$  can be influenced by node  $i$  where  $i$  and  $j$  are node at any layer [25].

The *threshold matrix* is represented by  $S = diag\{[\theta_1, \theta_2, \dots, \theta_n]\}$ . The diagonal elements of the matrix is the threshold of respective nodes. Let  $(M_{live})_{t=0}$  be the active set of nodes at  $t = 0$ , and the node changes from inactive state to active state in every steps.  $(M_{live})_t = \bigcup_{x=0}^t (M_{live})_x$ . The function continues until all nodes reaches active state.

#### 4 Entropy-Based Influence Minimization

In this section, we define the influence minimization in regular networks using the linear threshold model. We consider the public interest in the information and behavioural changes of users who receives the information. The physiological changes associated with emotion lead to a change in the energy level and the activity of each user. These emotional changes are analyzed through the LT model. The goal of the model is to minimize the influence of awful information. The number of activated nodes are needed to be minimum at the final stage of the information diffusion under node surveillance. We propose a survival model where the probability of a node  $v$ , that activate by the weighted sum of all probabilities of the previously activated nodes. The proposed algorithm blocks newly generated active nodes in the previous state,  $t_{n-1}$ .

##### 4.1 Problem Statement

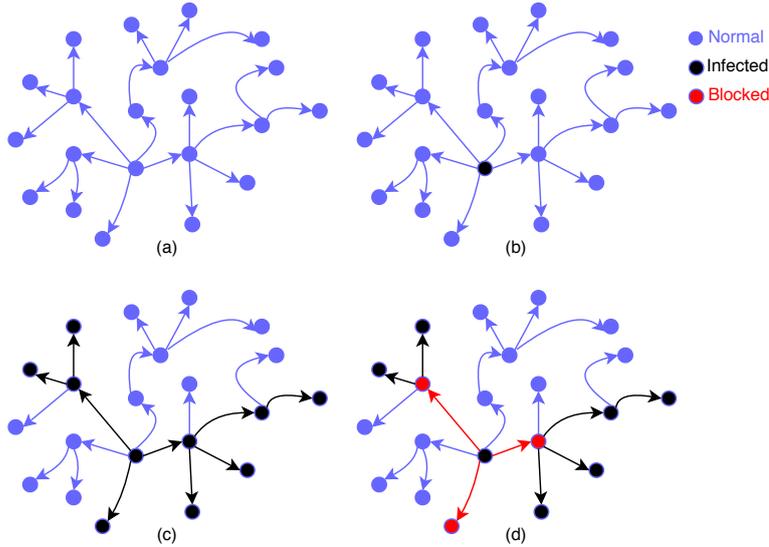
**Problem 1** Consider a directed multilayer graph,  $M = (V_M, E_M, V, L)$  where  $|V| = n$ , the total number of vertices and  $|L| = l$ , the total number of layers,  $V_M$  and  $E_M$  are the number of vertices and edges in each layer respectively. Let  $(M_{live})_0$  be the initial infected set of vertices, optimal subset of infected nodes as we expected as  $\sigma((M_{live})|V)$ . Our objective is to minimize the infection spread across the networks by removing  $m$  infected nodes. The final infected volume, denoted as  $(M_{live})^*$  is computed as,

$$(M_{live})^* = arg \text{Min}_{B \subseteq V, |B| \leq m} \sigma((M_{live})|V \setminus B) \quad (7)$$

Where  $\sigma((M_{live})|V \setminus B)$  is the number of infected nodes from  $V$  after removing  $B$ , set of immunized nodes. The  $(M_{live})$  is computed by greedy influence maximization function whereas  $B$  is computed based on entropy  $(\Delta E)$ .

The propagation of negative information causes the social contagion process [18]. As time increases, people's interest in such information tends to shrink with time. When a node receives a rumors, the energy level increases [15]. The

difference in energy level measure in terms of entropy and it is computed by the equation,  $\Delta E = cm_p \Delta T$ . Where  $\Delta T$  is change in heat energy,  $c$  is constant and  $m_p$  is *m-PageRank* that shows the energy distribution capability [9]. Since the node is activated, it transmits the information to its neighbours. The influence minimization achieves through either blocking the user for further transmission by annealing process or remove the link between infected nodes [3]. Remove a link cause isolate one group of nodes completely. The influence minimization process depicted in Fig. 2, which helps to understand well. The process of



**Fig. 2** The illustration of influence minimization. (a) a regular graph, (b) a node is infected, (c) information diffuse across networks and (d) blocking infected articulation nodes

removal of an edge possible only when the infected node is an articulation node. If the blocked time exceeds a threshold, the user either leaves away from the social networks or give intimation about the status. The system has to unblock or retain the link after a certain time-stamp. The time-stamp depends on how long the negative information lives in the system. The tolerance for latency is a major issue in the influence minimization problem.

#### 4.1.1 Solution Approach

The proposed influence minimization is a *greedy algorithm* where we find the optimal solution, either to maximize the threshold or to minimize it. The algorithm works better in the LT model, epidemic models. The degree centrality is the fundamental metric to find the most significant nodes and suits only for simple networks. Since degree centrality is a basic one and not suitable for multilayer networks. We select an appropriate metric, *m-PageRank*, for study-

ing the information diffusion. The formation of seed sets and malicious nodes depends on the propagation probability.

The algorithm picks an edge randomly which has a weight, the weight function,  $p : E_m \rightarrow [0, 1]$ . Let  $(i, j)$  be the *live-edge* with weight  $p_{ij}$ , all *live-edge* form a subset  $E_{live}$ . Each node that connects edges in the subset  $E_{live}$  has a threshold, let threshold of  $j$  be  $\theta_j$ . The node,  $j$ , is infected if  $p_{ij} \geq \theta_j$ . Let  $B$  is the subset of immunized nodes which are to be blocked for a duration,  $\Delta t$ . Let  $\sigma((M_{live})|B)$  be the expected number of infected node that  $(M_{live})$  can infect under LT mode given immunized set  $B$ . The e-IM problem aims to select subset of immunized nodes  $B$  with size equals to size of seed set. The e-IM function minimize the  $\sigma(M_{live}|\emptyset) - \sigma(M_{live}|B)$ .

---

**Algorithm 1** Algorithm for Influence minimization  $-(m-k)$  policy and  $(m-k)$ - $p$  policy

---

**Require:**  $M_{live}$  : Live nodes;  $m = |M_{live}|$ ;  $B$ : set of Blocked nodes:  $k$ :Seed;

**Ensure:**  $M_{live}^*$

- 1: **procedure** E-IM( $M_{live}, B, m, k$ )
- 2:     Initialize  $B \leftarrow \emptyset$ ;  $P \leftarrow \emptyset$ ;
- 3:     **for**  $w \leftarrow 1$  to  $n$  **do**
- 4:         **for**  $i \leftarrow 1$  to  $k$  **do**
- 5:             \ \*checking the threshold value for each node\*\
- 6:             **if**  $\sigma((M_{live} \setminus V - B_{i-1} \cup w - \sigma(M_{live} \setminus V - M_{live}_{(i-1)}))) < \Delta$  **then**
- 7:                  $\Delta E_w \leftarrow \text{Max}((\text{argMax}_{(w \in V-B)} \sigma(B \cup \{w\} - \sigma(B)), 1))$ ; \ \*calculate the entropy\*\
- 8:              $B \leftarrow B \cup \{w\}$ ;
- 9:             **for**  $j \leftarrow 1$  to  $m$  **do**
- 10:                  $M_{live}^* \leftarrow \text{argMax}(\sigma(M_{live}|V) - \sigma(M_{live}|V \setminus B))$  \ \*maximizing the influence\*\
- 11:             **while**  $j \neq n$  **do**
- 12:                 \ \*construct the DFS tree\*\
- 13:                 construct a *DFS-Tree*
- 14:                 consider an edge  $(u, v) \in E_{live}$
- 15:                 continue;
- 16:             **for**  $j \leftarrow 1$  to  $m$  **do**
- 17:                 **if**  $(v, p) \cup (p, u) \notin E$  **then**
- 18:                      $M_{live}^* \leftarrow \text{argMin}((\sigma(M_{live}|V) - \sigma(B|V)) - \sigma(P|V \setminus B))$  \ \*minimize the number of node influenced\*\
- 19:             Return( $M_{live}^*$ ) \ \*final list of live node\*\
- 20: **End procedure**

---

The algorithm for minimizing the influence spread by maintaining both policies are depicted in Algorithm 1. The two classical centrality metrics used as the basis for the function to maximize the spread and form a subset of nodes which are more influential to the given networks. The metrics described as follows:

- **Multiplex PageRank.** A node in a multilayer network resides at different layers, and these distinct positions constitute the rank of each node. The weight of the node in corresponding layers is aggregated with intra-layer

influence and votes [28]. Rank depends on the coupling relation between layers and characteristics of interconnections.

- **m-PageRank**. The rank of a node computes by considering the interconnection strength, not only from the same layer but from the neighbouring layers also. The rank of each layer accounted for the computation. It works based on a biased random walk and adds the impact of links from the other layers.

The performance of those metrics on the proposed algorithm analyzes through experiments and its analysis shown in section 6. Influence minimization problem further improved by  $((k-m)-p)$  policy. The problem of influence minimization takes in input as a multilayer social graph and produces a subset of nodes that block for minimizing the negative information spread across the networks. The algorithm finds live nodes from the network graph. A set of seed nodes form by using a centrality metric. The first policy state that the set of live nodes form by performing the set difference operation between the set of live nodes and seed.

$$M_{live}^* = M_{live} - B \quad (8)$$

The algorithm formulated for influence minimization based on the energy level is illustrated in the Algorithm 1 and it's time complexity is  $O(m^2k)$ . The notation,  $m$  is the cardinality of the live nodes,  $k$  is the size of the seed set. The algorithm has  $m$  iteration in the worst case.

The influence minimization has further optimized by second policy, where the articulation points have to remove from the set of live nodes. The subset of blocked nodes to be formed by the intersection of live nodes and seed by using the Algorithm 1. Further, optimized by the union of blocked nodes with a set of articulation nodes, so that a complete minimization of spread across the networks.

The procedure for the  $((k-m)-p)$  policy state as follows, during the first stage of the algorithm, construct a DFS tree and choose a random node  $u$ . Find any cyclic path exists to reach  $u$  through  $p$  selected nodes. If there is any path that exists, the node  $u$  adds to a previously defined set of articulation nodes,  $|\{P\}| = p$ . The process continues to discover all the nodes in the live graph.

## 5 Experiment and Validation

The competency of our proposed  $e-IM$  algorithm is validated through experiments. We have developed a python based simulator, where we used Python 3.6.0 and NetworkX 2.4 [14]. The proposed methodology compares with major algorithms for showing effectiveness. Series of experiments carried out using three real networks, such as NetScience [22], Hep-Th [16], and Facebook [29], that are used for social network analysis commonly. Brief detail about the dataset shown in Table 2. The results show how our proposed algorithm and policies are most suitable for the problem of influence minimization.

**Table 2** Statistics of real datasets.

Dataset	$Nodes(N)$	$Edges(E)$	$d_{avg}$	$d_{max}$
NetScience	1461	2745	3.75	34
Hep-Th	7610	15751	2.07	28
Facebook	46952	876993	2.696	37.35

The proposed method is compared with two of the state-of-the-art algorithms, DRIMUX [33] and DGMT [30].

**DRIMUX:** It is a rumor spreading minimization model called dynamic rumor influence minimization with user experience. It reduces the rumor propagation by blocking a certain subset of nodes in the network. if the block period exceeds the time limit, affect the utility of the whole the system DRIMUX considers the characteristics of the rumor and its experience of the user.

**DGMT:** It applies a linear restriction on seed set and find the optimal solution for minimizing diffusion using an integer linear programming problem.

### Performance Parameter

We want to block  $k$  susceptible nodes from the live nodes, where  $k$  nodes are most central as per  $m$ -PageRank ranking metric. Set of seed nodes ( $k = |seed|$ ) are to be removed from set of live nodes,  $m = |M_{live}|$ . Let  $p$  is the set of nodes which connects two or more connected components, where  $p \leq k$ . The optimal set of live nodes is to be formed by deducting  $p$  nodes to minimize the spread rate by isolating graph components.

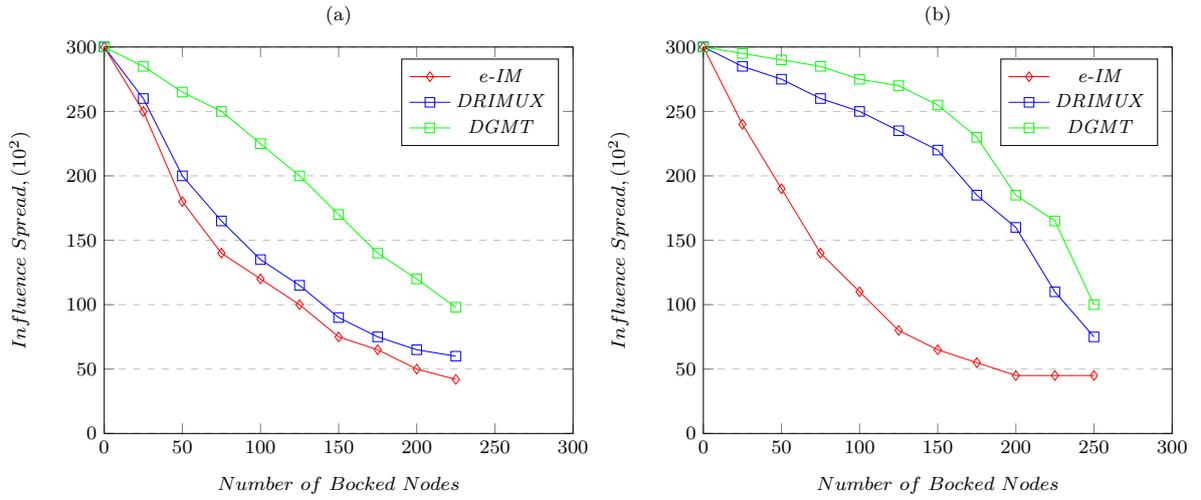
$$\text{Minimization Score}(\mu) = \frac{|\{seed\} \cup \{connected nodes\}|}{|M_{live}|}$$

$$\mu = \frac{|\{blocked nodes\}|}{|M_{live}|}$$

## 6 Result

The evaluation of the performance of our influence minimization policies on live networks described in Section 5. Let  $k$  be fixed to a constant, ( $k=50, k=100$ ), and compute number of nodes has been blocked while completing iteration on Facebook. We analyzed such that how many nodes has reached the negative information. Compare with DRIMUX and DGMT. The statistical relationship between blocked nodes and live nodes for Facebook are shown in the Fig. 3.

The experiments repeat for other networks by using other centrality metrics, DRIMUX, and DGMT. It found that as the blocked nodes increases, the number of nodes who received negative information gets decreased. The algorithm gives a more optimized solution and is found to be minimal spread while applying the  $((k-m)-p)$ , the second policy. Fig. 4 shows the correlation between blocked nodes and influence spread on Hep-Th and Netscience.



**Fig. 3** The graph shows the relationship between the number of blocked nodes and the number of nodes where influence spread. Set of seed ( $k = 50$ ) is a subset of blocked nodes and it is formed using metrics such as (a)  $m$ -PageRank and (b) Multiplex PageRank. As shown that the higher the value of the number of blocked nodes, the spread decreased. Our proposed policies gain a significant advantage

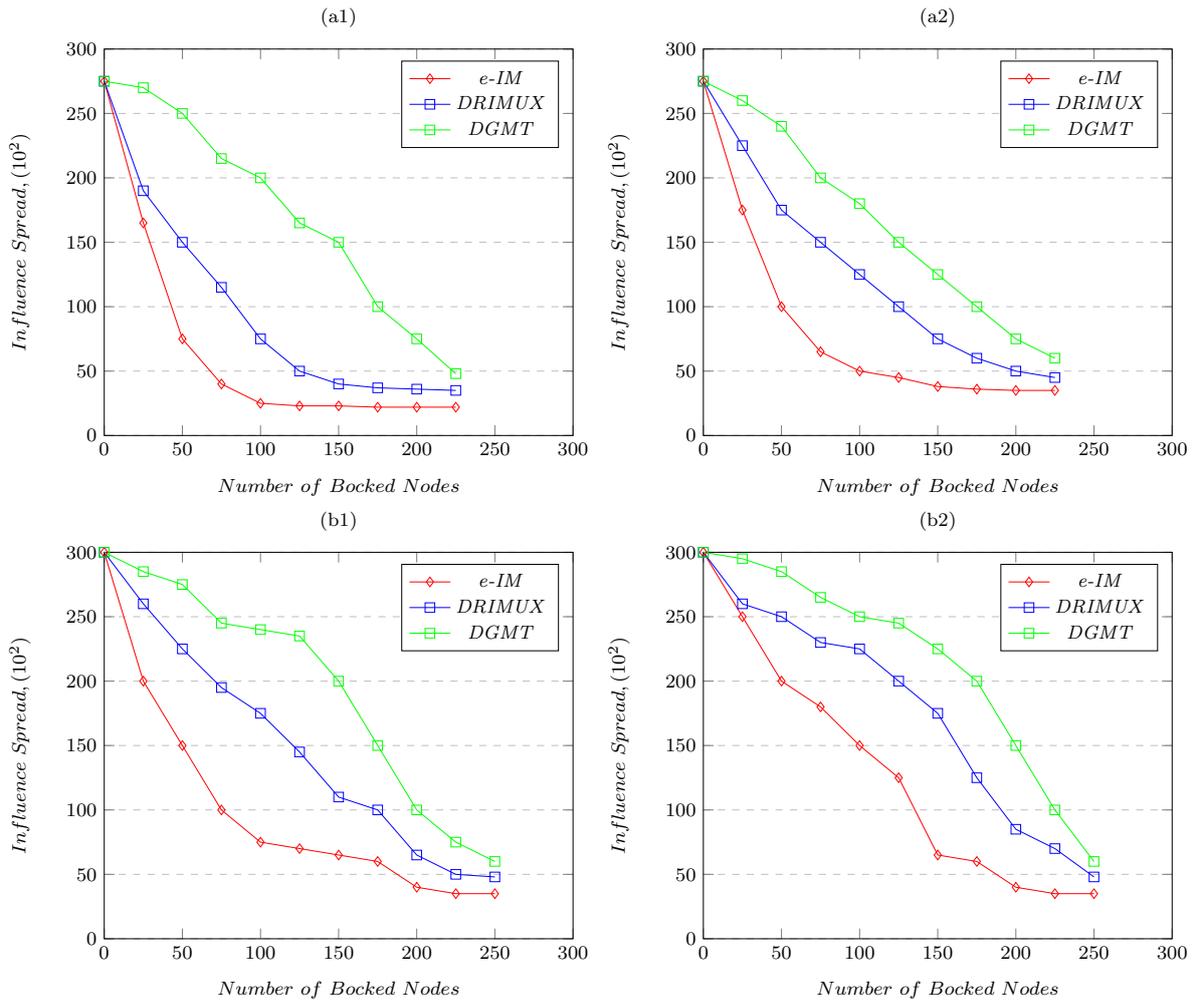
The metric that compare the influence minimization of various algorithm is made possible with help of minimization factor ( $\mu$ ). It is defined as the ratio of number of blocked nodes to the number of live nodes. The higher the value for  $\mu$ , the better the minimization capability for the algorithm. The comparison between  $e$ -IM and others are shown in the Table 3.

**Table 3** The value of  $\mu$  for influence minimization algorithm on real network under consideration.

Dataset	$e$ -IM	DRIMUX	DGMT
NetScience	10.22	6.43	4.69
Hep-Th	7.14	5.21	4.17
Facebook	5.35	3.75	2.29

## 7 Conclusion

In this article, we formulated two policies for developing an algorithm capable to minimize the influence spread across the network. We have identified the most central nodes in the network through a leading ranking metric,  $m$ -PageRank. By blocking or removing such nodes make the network capable to minimize the information spreading. The algorithm further optimized by adding an another policy which isolate nodes by infecting them with virus. A



**Fig. 4** The graph shows the relation between number of blocked nodes and number of nodes where influence spread, where  $k = 50$ . (a1) NetScience using *m-PageRank*, (a2) NetScience using Multiplex PageRank, (b1) Hep-Th using *m-PageRank*, and (b2) Hep-Th using Multiplex PageRank

python based simulator has developed and experimented with real networks. The result shows that the proposed method outperform compared with state-of-the-art. The future enhancement for the work can be done by scaling it for present pandemic environment. It can be modified for implementing an effective method for isolating people from any kind of viral diseases.

## Availability of data and materials

The experimentation has been done with the dataset *NetScience* at <http://konect.cc/networks/dimacs10netscience/>, *Hep-Th* at <https://snap.stanford.edu/data/citHepTh.html>, and *Facebook* at <http://konect.cc/networks/facebook-wosnwall/>. The validation of the experiments as a subpart of the multiple spreading process and is available at <https://github.com/JoCherian/>

## References

1. Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. *Reviews of modern physics* **74**(1), 47 (2002)
2. Archambeau, C., Lakshminarayanan, B., Bouchard, G.: Latent ibp compound dirichlet allocation. *IEEE transactions on pattern analysis and machine intelligence* **37**(2), 321–333 (2014)
3. Bertsimas, D., Tsitsiklis, J., et al.: Simulated annealing. *Statistical science* **8**(1), 10–15 (1993)
4. Bharathi, S., Kempe, D., Salek, M.: Competitive influence maximization in social networks. In: *International workshop on web and internet economics*, pp. 306–311. Springer (2007)
5. Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C.I., Gómez-Gardenes, J., Romance, M., Sendina-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* **544**(1), 1–122 (2014)
6. Borodin, A., Filmus, Y., Oren, J.: Threshold models for competitive influence in social networks. In: *International workshop on internet and network economics*, pp. 539–550. Springer (2010)
7. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 199–208 (2009)
8. Cheriyan, J., Sajeev, G.: Spreadmax: a scalable cascading model for influence maximization in social networks. In: *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1290–1296. IEEE (2018)
9. Cheriyan, J., Sajeev, G.P.: An improved pagerank algorithm for multilayer networks. In: *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, pp. 1–6. IEEE (2020)
10. Cozzo, E., De Arruda, G.F., Rodrigues, F.A., Moreno, Y.: *Multiplex networks: basic formalism and structural properties*. Springer (2018)
11. De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivela, M., Moreno, Y., Porter, M.A., Gómez, S., Arenas, A.: Mathematical formulation of multilayer networks. *Physical Review X* **3**(4), 041022 (2013)
12. Glauber, R.J.: Time-dependent statistics of the ising model. *Journal of mathematical physics* **4**(2), 294–307 (1963)
13. Gomez, S., Diaz-Guilera, A., Gomez-Gardenes, J., Perez-Vicente, C.J., Moreno, Y., Arenas, A.: Diffusion dynamics on multiplex networks. *Physical review letters* **110**(2), 028701 (2013)
14. Hagberg, A., Schult, D., Swart, P., Conway, D., Séguin-Charbonneau, L., Ellison, C., Edwards, B., Torrents, J.: *Networkx. high productivity software for complex networks*. <https://networkx.lanl.gov/wiki> (2013)
15. Han, S., Zhuang, F., He, Q., Shi, Z., Ao, X.: Energy model for rumor propagation on social networks. *Physica A: Statistical Mechanics and its Applications* **394**, 99–109 (2014)
16. <https://snap.stanford.edu/data/citHepTh.html>
17. Jin, L., Chen, Y., Hui, P., Ding, C., Wang, T., Vasilakos, A.V., Deng, B., Li, X.: Albatross sampling: robust and effective hybrid vertex sampling for social graphs. In: *Proceedings of the 3rd ACM international workshop on MobiArch*, pp. 11–16 (2011)

18. Kawachi, K., Seki, M., Yoshida, H., Otake, Y., Warashina, K., Ueda, H.: A rumor transmission model with various contact interactions. *Journal of theoretical biology* **253**(1), 55–60 (2008)
19. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146 (2003)
20. Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of complex networks* **2**(3), 203–271 (2014)
21. Medya, S., Silva, A., Singh, A.: Influence minimization under budget and matroid constraints: Extended version. *arXiv preprint arXiv:1901.02156* (2019)
22. [http://konect.cc/networks/dimacs10\\_netscience/](http://konect.cc/networks/dimacs10_netscience/):
23. Newman, M.: *Networks: an introduction*. Oxford university press (2010)
24. Newman, M.E.: The structure and function of complex networks. *SIAM review* **45**(2), 167–256 (2003)
25. Rahimkhani, K., Aleahmad, A., Rahgozar, M., Moeini, A.: A fast algorithm for finding most influential people based on the linear threshold model. *Expert Systems with Applications* **42**(3), 1353–1361 (2015)
26. Su, X., Wang, M., Fan, J.: Maximum likelihood regression trees. *Journal of Computational and Graphical Statistics* **13**(3), 586–598 (2004)
27. Tomasini, M.: An introduction to multilayer networks. *BioComplex Laboratory, Florida Institute of Technology, Melbourne, USA* pp. 1–14 (2015)
28. Tu, X., Jiang, G.P., Song, Y., Zhang, X.: Novel multiplex pagerank in multilayer networks. *IEEE Access* **6**, 12530–12538 (2018)
29. [http://konect.cc/networks/facebook-wosn\\_wall/](http://konect.cc/networks/facebook-wosn_wall/):
30. Wang, B., Chen, G., Fu, L., Song, L., Wang, X.: Drimux: Dynamic rumor influence minimization with user experience in social networks. *IEEE Transactions on Knowledge and Data Engineering* **29**(10), 2168–2181 (2017)
31. Wang, X., Deng, K., Li, J., Yu, J.X., Jensen, C.S., Yang, X.: Targeted influence minimization in social networks. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 689–700. Springer (2018)
32. Yang, L., Giua, A., Li, Z.: Minimizing the influence propagation in social networks for linear threshold models. *IFAC-PapersOnLine* **50**(1), 14465–14470 (2017)
33. Yang, L., Li, Z., Giua, A.: Influence minimization in linear threshold networks. *Automatica* **100**, 10–16 (2019)
34. Yao, Q., Shi, R., Zhou, C., Wang, P., Guo, L.: Topic-aware social influence minimization. In: *Proceedings of the 24th International Conference on World Wide Web*, pp. 139–140 (2015)

# Figures

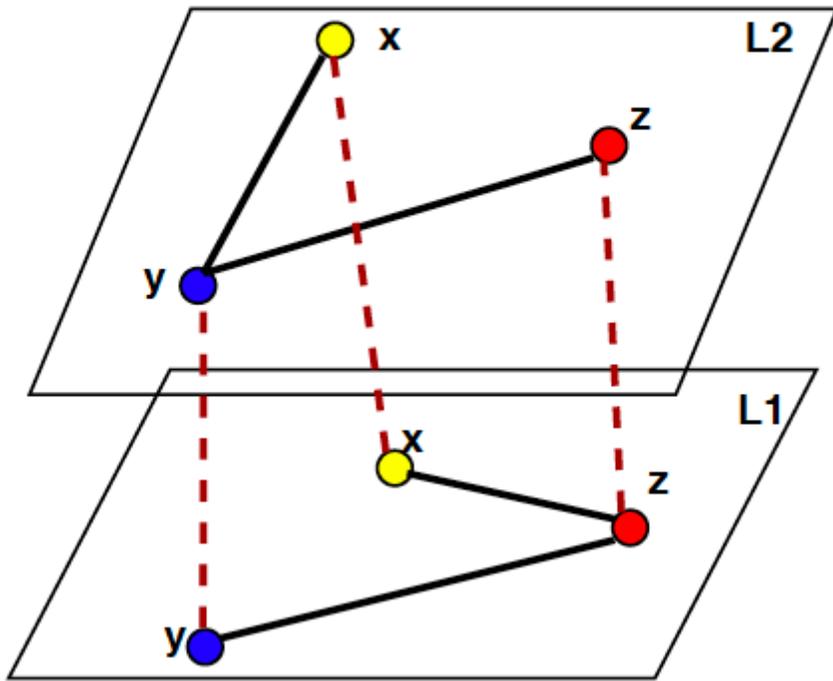


Figure 1

A multiplex network with  $V = \{x, y, z\}$  and  $L = \{L1, L2\}$

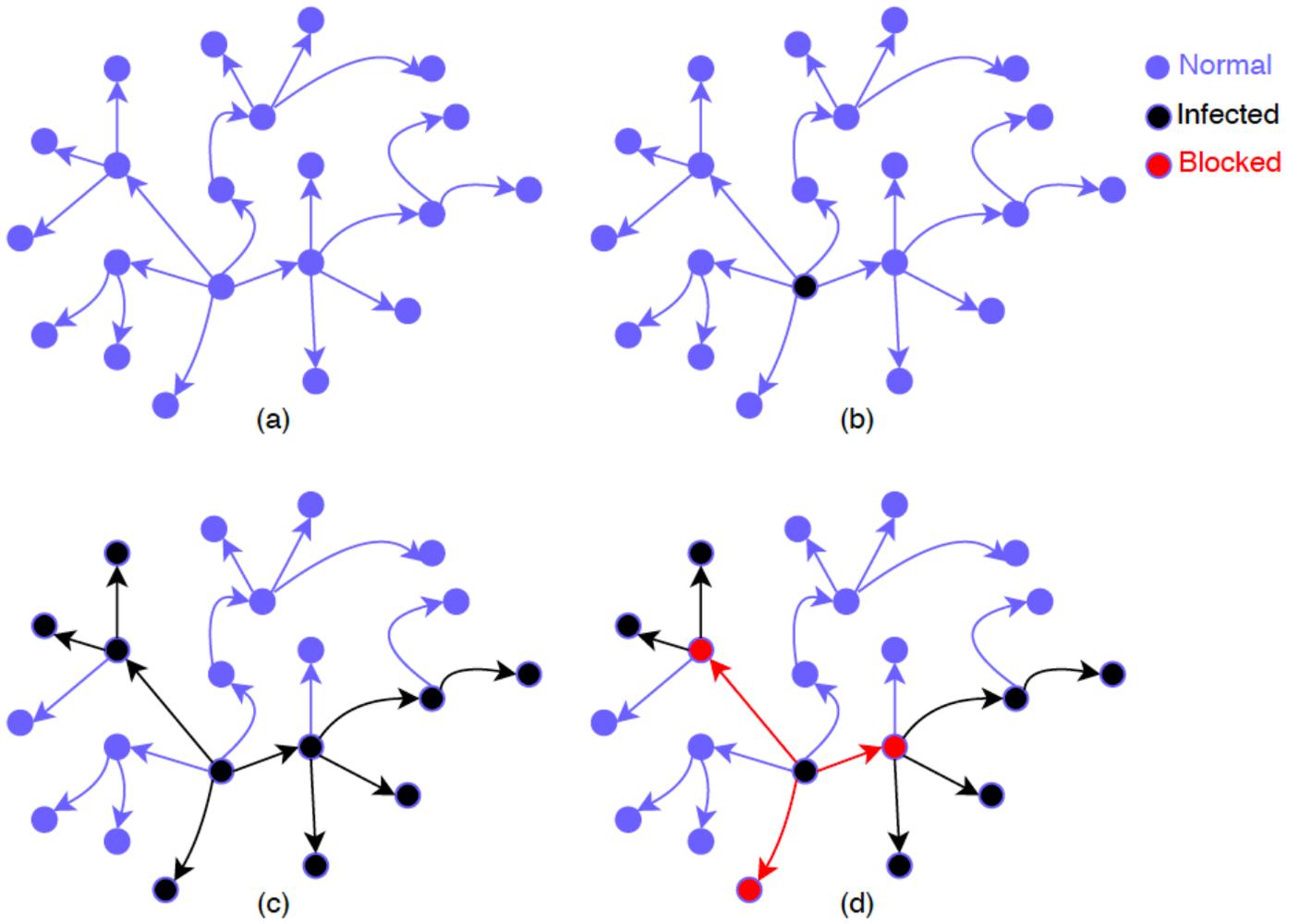
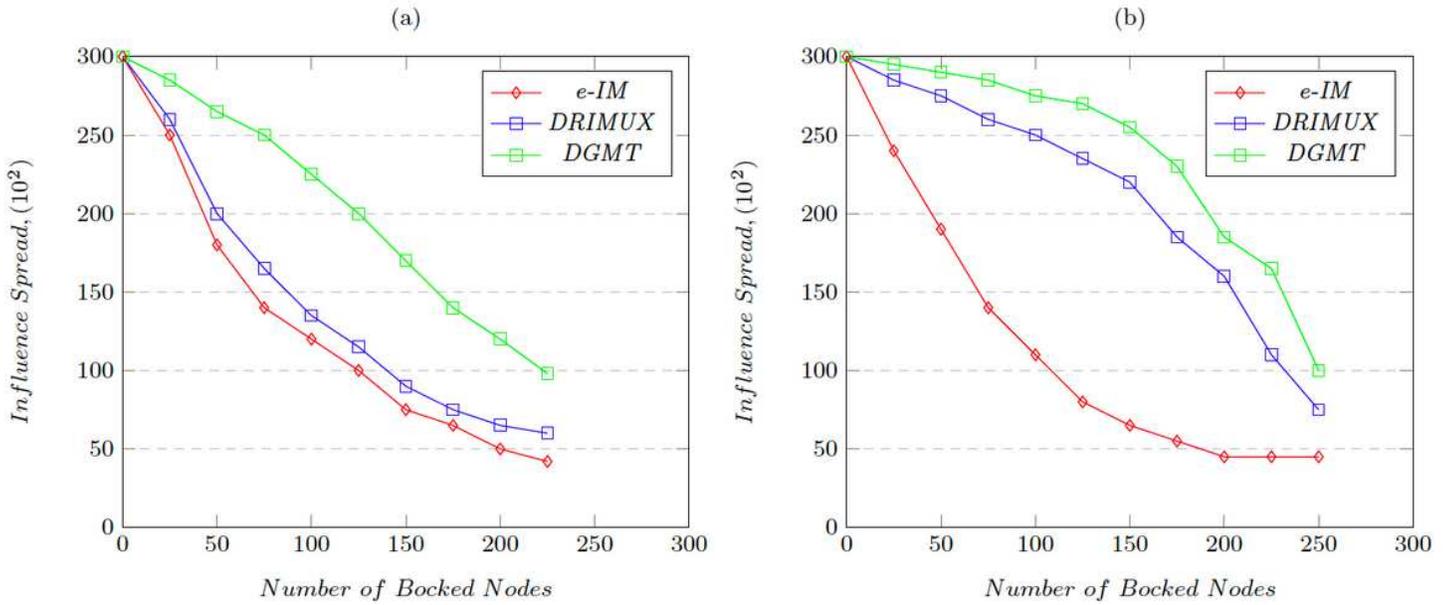


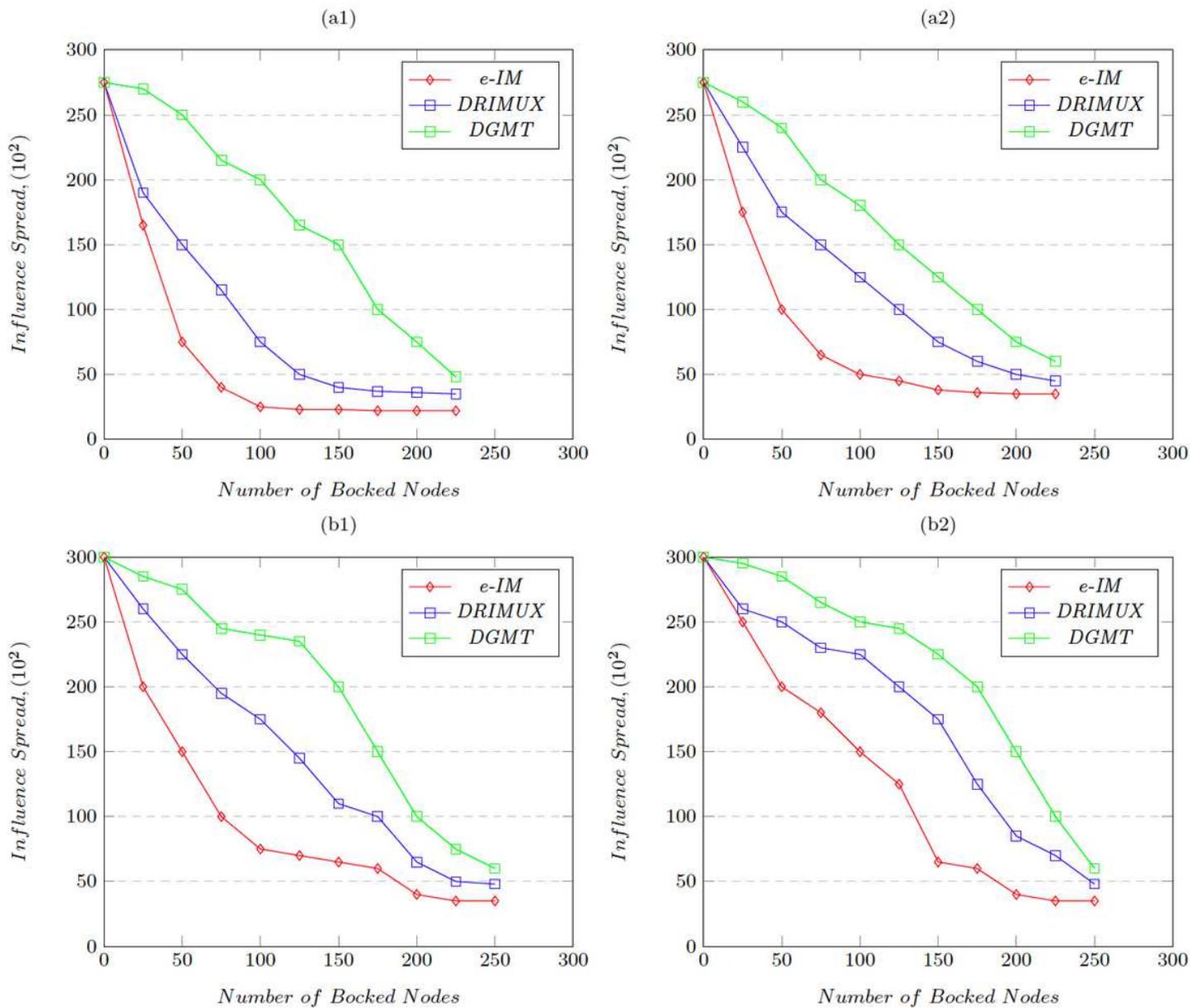
Figure 2

The illustration of influence minimization. (a) a regular graph, (b) a node is infected, (c) information diffuses across networks and (d) blocking infected articulation nodes



**Figure 3**

The graph shows the relationship between the number of blocked nodes and the number of nodes where influence spread. Set of seed ( $k = 50$ ) is a subset of blocked nodes and it is formed using metrics such as (a) m-PageRank and (b) Multiplex PageRank. As shown that the higher the value of the number of blocked nodes, the spread decreased. Our proposed policies gain a significant advantage



**Figure 4**

The graph shows the relation between number of blocked nodes and number of nodes where influence spread, where  $k = 50$ . (a1) NetScience using m-PageRank, (a2) NetScience using Multiplex PageRank, (b1) Hep-Th using m-PageRank, and (b2) Hep-Th using Multiplex PageRank