Applied Mathematics and Computation 320 (2018) 512–523

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

# Identification of influential spreaders based on classified neighbors in real-world complex networks

Chao Li[a,b], Li Wang[a,b,*], Shiwen Sun[a,b], Chengyi Xia[a,b,*]

[a] Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin University of Technology, Tianjin 300384, PR China

[b] Key Laboratory of Computer Vision and System (Ministry of Education), Tianjin University of Technology, Tianjin 300384, PR China

ARTICLE INFO

*Keywords:*
Influential spreaders
Identification algorithms
Classified neighbors
Complex networks

ABSTRACT

Identifying the influential spreaders in complex network is a very important topic, which is conducive to deeply understanding the role of nodes in the information diffusion and epidemic spreading among a population. To this end, in this paper, we propose a novel classified neighbors algorithm to quantify the nodal spreading capability and further to differentiate the influence of various nodes. Here, we believe that the contribution of different neighbors to their focal node is different, and then classify the neighbors of the focal node according to the removal order of the neighbor in the process of *k*-shell decomposition. By assigning different weights for each class of neighbors and summing up the neighbors' contributions, the spreading capacity of the focal node can be accurately characterized. Through extensive simulation experiments over 9 real-world networks, the weight distribution of different types of neighbors has been optimized, and the results strongly indicate that the current algorithm has the higher ranking accuracy and differentiation extent when compared to other algorithms, such as degree centrality, *k*-shell decomposition method and mixed degree decomposition approach. Current results can help to greatly reduce the cost of sales promotion, considerably suppress the rumor dissemination and effectively control the outbreak of epidemics within many real-world systems.



## 1. Introduction

Over the past two decades, various dynamics taking place upon complex networks [1–10] receive a great deal of concern, such as synchronization, consensus, evolutionary game, epidemics and opinion dynamics. Among them, the spreading process over networks [11] is an important subject in the field of physics, chemistry, public hygienics, biology and even sociology. As an example, influenza spreading, rumor diffusion, cascading failure, or information dissemination can be all incorporated into the range of spreading process. To be particularly worth mentioning, the epidemic spreading in complex networks has been extensively investigated and many studies have clearly indicated that this process is strongly correlated with the topology of networks [11,12]. Among them, Pastor-Satorras and Vespignani [13] defined a dynamical model for the epidemic spreading in scale-free networks, and found a striking phenomenon which declared the absence of an epidemic threshold and its associated critical behavior under the thermodynamic limit. Starting from this work, many models have

* Corresponding author at: Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin University of Technology, Tianjin 300384, PR China.

*E-mail addresses:* wltjut08@126.com, xialooking@163.com (C. Xia).

https://doi.org/10.1016/j.amc.2017.10.001
0096-3003/

been developed to analyze the spreading properties on networks and then to increase the spreading threshold for a specific disease [14–23]. As a further step, several efficient immunization schemes [24–28] have also been put forward to inhibit the outbreaks of infectious diseases.

Although large quantities of previous works try to macroscopically understand and control the spreading process, in the very recent years more and more attention has been paid to microscopically studying the individual spreading capability, which characterizes the number of nodes covered within a specified process when a single agent initiates an epidemic [29–32]. In fact, knowing the spreading capability for each node is vital for us to devise efficient methods to control the guidance of public opinion [33,34], promote the popularity of novel products [35–37] and even refrain the outbreaks of diseases within the population [38,39]. In addition, it will be also beneficial to search for the spreading origin (i.e., initial spreader) of a certain disease or information [40].

At present, several classic topology indicators can be used to quantitatively characterize the individual spreading capabilities [41–52], such as degree, closeness, betweenness,clustering coefficient, Katz centrality and so on. Despite the intuition that the most connected nodes (hubs) or nodes with high betweenness centrality are thought to be very influential spreaders inside networks, many works have also revealed that the real-world spreading scenarios cannot well agree with these ideas. Thus, exploring the novel index to denote the individual spreading capacity becomes an intriguing topic. Among them, the $k$-shell decomposition algorithm [48] takes advantage of the debarking method to rank the nodes. In this method, the algorithm starts by iteratively removing all nodes with degree 1 until no such these nodes remain, and we assign their $k$-shell value to be 1. In a similar fashion, we will obtain the 2-shell nodes by recursively removing the remaining nodes with degree 2, and this procedure continues until all nodes are assigned to a $k$-shell value. It is obvious that the nodes with high $k$-shell value will tend to locate at the center of the network, and hence the covered range originating from these nodes is likely to be larger, which means the higher spreading capability. Based on some real infectious diseases, this method was found to perform better than other methods in identifying the influential spreaders.

However, when the $k$-shell method is used to decompose the networks, we only consider the remaining links among the existent nodes (named as the residual links, correspondingly the concept of residual degree), but often neglect the links connecting to the removed nodes (called the exhausted links, accordingly exhausted degree). In order to hold the same spreading capability, all nodes within the same core should have the identical number of exhausted links, at least a very small fluctuation in the exhausted degree. In reality, the exhausted degree often exhibits a very heterogeneous distribution in real networks. Thus, all nodes with the same residual degree will be assigned the same rank even though their exhausted degree may fluctuate much more. As an example, the $k$-shell method may designate the same $k$-shell value for many nodes in tree-like network or Barabási-Albert networks. Therefore, it is found that $k$-shell method is efficient in identifying the most influential spreader for infectious disease, but it is not too suitable to sort the nodes regarding the information or rumor propagation, and thus there exists much room for us to improve the quality of $k$-shell method since $k$-shell method denotes a very coarse-grained decomposition. For instance, Zeng and Zhang [52] propose a mixed degree decomposition (MDD) procedure to predict the spreading capability by combining the residual degree and exhausted degree, which can effectively reduce the degeneracy of the $k$-shell algorithm. Furthermore, Liu et al [53,54] take use of optimization algorithm to remove the core-like groups and redundant links inside the networks and then adopt the $k$-shell method to identify the true core within networks, and henceforth the performance of $k$-shell method has been further enhanced.

As mentioned in Ref. [52], different neighbors may play a distinct role in measuring the individual spreading capability (SC). Thus, in this paper, we propose a well-refined method to further classify the neighbors according to their relative sequence of being removed during the $k$-shell decomposition. Then, we will set a different weight for each class of neighbors to characterize the spreading capability, and further simulate the Susceptible-Infected-Removed (SIR) epidemic model on several typical real-world network data sets. It is evidenced that the proposed algorithm is better than degree centrality, $k$-shell, and MDD method regarding the performance of measuring the spreading capability. Meanwhile, the time complexity of our method is also largely reduced when compared to that of MDD algorithm.

The rest of this paper is structured as follows. In Section 2, we introduce the neighbor classification in detail, and then we briefly describe the SIR epidemic model and the corresponding real-world network data sets adopted in this paper. After that, we will carry out extensive numerical simulations upon 9 real-world networks and compare the present method and other typical algorithms in Section 3. Finally, we end this paper with some concluding remarks in Section 4.

## 2. Model and algorithm

### 2.1. Classified neighbors

In order to present the idea on the node assortment, we firstly look at a typical scenario shown in Fig. 1, where the spreading capability of node $d$ is usually different from those of nodes $a$, $b$ and $c$ although their $k$-shell value ($k_S$) is identical ($k_S = 1$). Here, node $d$ is removed after nodes $a$, $b$ and $c$ have been deleted from the network, and the coarse-grained $k$-shell decomposition does not consider this sequence. Analogously, in the $k_S = 2$ layer, nodes $j$ and $l$ are also different from nodes $k$, $m$ and $n$ even though their $k$-shell values are all 2. Therefore, it is necessary to take the removal sequence within the same $k_S$ layer into account and further classify the neighboring nodes for each node with the same $k_S$ values during the network decomposition. Our method starts from the aforementioned assortment mechanism regarding the nodes with the same $k_S$ value, and we will simultaneously record the $k$-shell value and the removing order for each node during the process

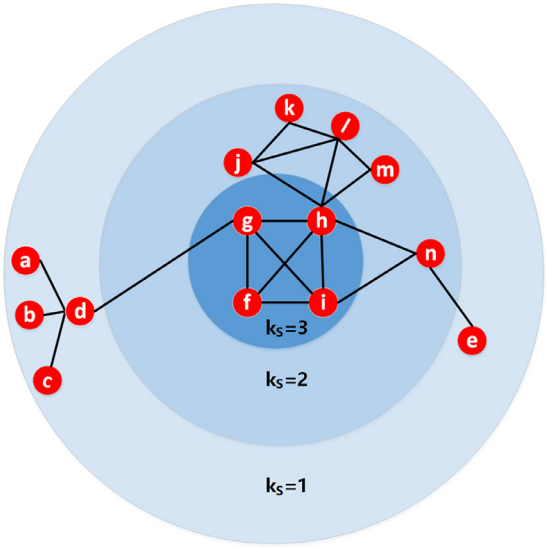


**Fig. 1.** Illustration of $k$-shell decomposition method, in which nodes with the low $k_S$ value lies in the peripheral layer and high $k$-shell value nodes are situated at the center of networks. Starting from nodes with degree $k = 1$, we assign $k_S = 1$ to them and remove them, repeat this process until all nodes have $k \geq 2$. Through this pruning procedure, we can obtain different $k_S$ layers in the whole network. On the basis of this method, we will record the order or sequence of nodes being removed, and propose the method of classified neighbors.

**Table 1**
The classification of neighbors for each node in Fig. 1.

| Node | Upper | Equ_upper | Equ_lower | Lower |
|---|---|---|---|---|
| ***a*** | – | *d* | – | – |
| ***b*** | – | *d* | – | – |
| ***c*** | – | *d* | – | – |
| ***d*** | *g* | – | *a,b,c* | – |
| ***e*** | *n* | – | – | – |
| ***f*** | – | *g,h,i* | – | – |
| ***g*** | – | *f,h,i* | – | *d* |
| ***h*** | – | *f,g,i* | – | *j,l,m,n* |
| ***i*** | – | *f,g,h* | – | *n* |
| ***j*** | *h* | *l* | *k* | – |
| ***k*** | – | *j,l* | – | – |
| ***l*** | *h* | *j* | *k,m* | – |
| ***m*** | *h* | *l* | – | – |
| ***n*** | *h,i* | – | – | *e* |

of network decomposition. As shown in Fig. 1, for any focal node, we will categorize its neighbors into 4 classes according to their $k_S$ values and the sequence of being removed as follows.

- Upper neighbors: their $k_S$ values are greater than the $k$-shell value of the focal node. For example, node $g$ is the upper neighbor of node $d$ in Fig. 1 (which is also used to illustrate the following 3 kinds of neighbors).
- Equal upper neighbors: their $k_S$ values are equal to the $k$-shell value of the focal node, but they are removed at the same time as the focal node or they are deleted from the network after the focal one has been removed. For instance, node $l$ is the equal upper neighbor of nodes $j$, $k$ and $m$.
- Equal lower neighbors: their $k_S$ values are equal to the $k$-shell value of the focal node, but their removal sequence is earlier, e.g., nodes $a$, $b$, $c$ are all the equal lower neighbors of node $d$.
- Lower neighbors: their $k_S$ values are less than the $k$-shell value of the focal node. As an example, node $d$ is the lower neighbor of node $g$.

Based on the current classifying method, all neighboring nodes in Fig. 1 can be summarized in Table 1. Meanwhile, the number of neighbors for above-mentioned 4 classes are denoted as $e^u$, $e^{eu}$, $e^{el}$ and $e^l$, respectively. Then, on the basis of classified neighbors (CN), we characterize the spreading capability of each node as follows

$$k_S^{CN} = \alpha \times e^u + \beta \times e^{eu} + \gamma \times e^{el} + \mu \times e^l \tag{1}$$

where $\alpha$, $\beta$, $\gamma$ and $\mu$ are tunable parameters which lie between 0 and 1, their specific setup will be discussed in Section 3.1. Henceforth, the algorithm based on classified neighbors to quantify the nodal spreading capability has been outlined in Table 2.

**Table 2**
The algorithm to evaluate the spreading capability using our classified neighbors method.

Algorithm 1

**Input :** A network **G** with **N** nodes and **E** edges; factor of upper neighbors $\boldsymbol{\alpha}$, factor of equ_upper neighbors $\boldsymbol{\beta}$, factor of equ_lower neighbors $\boldsymbol{\gamma}$, factor of lower neighbors $\boldsymbol{\mu}$
**Output:** the spreading capability $k_s^{CN}$ of each node in the network based on CN algorithm
**//Step1: decompose the network based on k-shell method and record the order of nodes being removed**
**1 :** $i = 1$;
**2 :** del_order = 1;// record nodes removed order
**3 :** $G'$ = copy of $G$;
**4 :** while (there exist residual nodes in $G'$)
**5 :** while (there exist nodes with degree i in the remaining $G'$)
**6 :** assign these nodes to $i$-shell;
**7 :** remove these nodes and set their removal order as del_order;
**8 :** del_order++;
**9 :** end
**10 :** $i$++;
**11:** end
**//Step2: classify the neighbors of each node based on the removal order gained in step1 and record each node's different neighbors as $e^u$, $e^{eu}$, $e^{el}$, $e^l$**
**12:** one node belongs to the focal's upper neighbor if its $k_S$ is larger
**13:** one node belongs to the focal's lower neighbor if its $k_S$ is smaller
**14:** one node belongs to the focal's equ_upper neighbor if its $k_S$ equals to the focal but its del_order is larger
**15:** one node belongs to the focal's equ_lower neighbor if its $k_S$ equals to the focal but its del_order is smaller
**//Step3: calculate the spreading capability $k_S^{CN}$ of each node based on step2**
**16:** $k_S^{CN} = \alpha \times e^u + \beta \times e^{eu} + \gamma \times e^{el} + \mu \times e^l$

**Table 3**
The algorithm to measure the individual spreading capability using SIR model.

Algorithm 2

**Input :** A network **G** with **N** nodes and **E** edges; the seed $n_{seed}$ is the only infected node at the initial step, infection probability $\boldsymbol{\lambda}$, recovery probability $\boldsymbol{\xi}$
**Output:** the total number ($N_R$) of nodes with the status $R$ in the network
**1:** Initialize the node status: $n_{seed}$=I, $n_{others}$=S
**2:** while (there exists the node whose status is $I$)
**3:** the infective nodes ($I$) infect their susceptible neighbors with the probability $\boldsymbol{\lambda}$ ($S \to I$), we just record the neighbors who get infected, but will not immediately update their status at this time step
**4:** the infective nodes recovered with the probability $\boldsymbol{\xi}$ ($I \to R$)
**5:** update the status of infected neighbors at **step 3**
**6:** end

### *2.2. SIR model and its simulation algorithm*

The susceptible-infected-recovered (SIR) model is a classical epidemic compartment model which involves three different states:

- Susceptible state: the healthy individuals lie in this state.
- Infected state: the individuals have been infected and have some chances to spread the disease or infect others.
- Recovered state: infective agents have been cured and will not be infected again, meanwhile they will not infect others, either.

In order to measure the spreading capability of each node, only one node will be initialized as an infective seed and other nodes be set as susceptible ones at each independent run. At each time step, an infective node will try to infect his neighbors with the probability $\lambda$, meanwhile he will be cured at the rate $\xi$ and enter into the recovered state. Once the infected agents are extinct (i.e., there are only $S$ and $R$-state ones), the spreading process will be stopped and the system will arrive at the stationary state. At this instant, we will record the total number of recovered individuals as the spreading capability of the initial infective seed, and the detailed computing procedure has been summarized in Table 3. In this paper, the SIR model will be considered as a standard benchmark to compare the performance of various mechanisms when we discuss the node spreading capability.

During the simulations, to reduce the impact of random noise, the final results will be averaged over 500 independent runs. Without loss of generality, the recovering rate $\xi$ is set to be 1.0, that is, each infective agent can only have one chance to infect his neighbors and become recovered at next time step. In addition, we can utilize the mean-field theory to derive

**Table 4**
The topological properties of 9 real-world networks used in the numerical simulations.

| Network | $N$ | $E$ | $<k>$ | $k_{max}$ | $ks_{max}$ | $C$ | $r$ | $H_k$ | $\lambda_c$ | $\lambda$ | $N_C$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Blogs | 3982 | 6803 | 3.4 | 189 | 7 | 0.284 | −0.133 | 4.038 | 0.078 | 0.09 | 11 |
| Netsci | 379 | 914 | 4.8 | 34 | 8 | 0.741 | −0.082 | 1.663 | 0.142 | 0.15 | 9 |
| Router | 5022 | 6258 | 2.5 | 106 | 7 | 0.012 | −0.138 | 5.503 | 0.079 | 0.09 | 26 |
| Power | 4941 | 6594 | 2.7 | 19 | 5 | 0.08 | 0.0035 | 1.45 | 0.348 | 0.36 | 12 |
| PGP | 10,680 | 24,316 | 4.6 | 205 | 31 | 0.266 | 0.238 | 4.146 | 0.056 | 0.07 | 41 |
| Email | 1133 | 5451 | 9.6 | 71 | 11 | 0.220 | 0.078 | 1.942 | 0.057 | 0.08 | 12 |
| Email contact | 12,625 | 20,362 | 3.2 | 576 | 23 | 0.109 | −0.387 | 34.249 | 0.009 | 0.03 | 41 |
| Hamster | 2000 | 16,097 | 16.1 | 273 | 24 | 0.54 | 0.023 | 2.719 | 0.023 | 0.04 | 25 |
| Astro | 14,845 | 119,652 | 16.1 | 360 | 56 | 0.670 | 0.02 | 2.820 | 0.02 | 0.03 | 57 |

the epidemic threshold [54–56] regarding the infection rate $\lambda$ in the networked population as follows,

$$\lambda_C = \frac{<k>}{<k^2> - <k>} \tag{2}$$

where $<k>$ and $<k^2>$ denote the first and second moment of the whole network, respectively. For the sake of being fully propagated, $\lambda$ is usually set to be $\lambda > \lambda_C$, but nearly all of nodes will be infected if the rate $\lambda$ is sufficiently high. In general, we set the rate $\lambda$ to be a little greater than $\lambda_C$ so as to clearly describe the spreading capability of nodes within a given network.

### 2.3. Real-world network data sets

In this paper, we select 9 typical real-world networks to perform extensive simulations and then compare the results between our model and other methods, and these networks include as follows.

- Blogs: Communication network of Blogs [57].
- NetSci: Co-authorship network of scientist [58].
- Router: Internet topology at the router level [59].
- Power: Western states power grid [60].
- PGP: Confidential communication network using the Pretty Good Privacy (PGP) encryption algorithm [61].
- Email: Email contact topology at computer science department of University College London (UCL) [62].
- Email contact: Email contact topology of University at Roviria and Virgili (URV) in Spain [48].
- Hamster: Relationship topology of users at Hamster.com [54].
- Astro: Collaboration network of Astrophysics scientists [63].

For these real-world networks, several statistical indicators regarding their topology can be summarized in Table 4, which include the total number of nodes ($N$), the total number of edges ($E$), the average degree or the first moment of network ($<k>$), the maximum degree ($k_{max}$), maximum $k$-shell value ($k_{Smax}$), clustering coefficient ($C$), degree assortativity ($r$), degree heterogeneity ($H_k = \frac{<k^2>}{<k>^2}$), epidemic threshold ($\lambda_C$) and so on. Moreover, the specific infection rate ($\lambda$) for each network in our simulations and the number of nodes ($N_C$) within the highest $k$-shell layers are also listed in Table 4.

## 3. Simulation results

### 3.1. Weight assignment of classified neighbors

Based on the aforementioned idea of node classification, the neighbors of the focal node can be divided into four classes. According to Eq. (1), when we measure the spreading capability regarding the classified neighbors, their corresponding weight has been assigned as $\alpha$, $\beta$, $\gamma$ and $\mu$, respectively. Generally, the closer to the network core the neighboring nodes, the more their contribution to the focal node spreading capability. To this end, we assume that the weight relationship among 4 classes of neighbors can be ranked as $\alpha > \beta > \gamma > \mu$.

To validate the correctness of the assumption about the above weight ranking, we can consider all possible weight parameter setups and calculate their $k_S^{CN}$ ranking under any combination of these weights. After that, we need to compare the average rankings with the ones obtained under the standard SIR model to check whether these two rankings are consistent.

Furthermore, to simplify the discussed problems, we temporarily combine the equal upper neighbors and equal lower neighbors as equal neighbors and let $e^{eq} = e^{eu} + e^{el}$ so that $e^{eq}$ denotes the total number of neighbors with the same $k$-shell value as the focal node, meanwhile $\delta$ is used to stand for their weight and hence the nodal spreading capability can be rewritten as,

$$k_S^{CN} = \alpha \times e^u + \delta \times e^{eq} + \mu \times e^l \tag{3}$$

**Table 5**
The average spreading ability of different neighbors in the classified neighbors method.

| Network | $U_{SC}$ | $EU_{SC}$ | $EL_{SC}$ | $L_{SC}$ | $U_{nm}$ | $EU_{nm}$ | $EL_{nm}$ | $L_{nm}$ |
|---|---|---|---|---|---|---|---|---|
| Blogs | 7.945 | 4.165 | 3.641 | 1.861 | 0.451 | 0.236 | 0.207 | 0.106 |
| Netsci | 7.166 | 4.308 | 2.971 | 3.005 | 0.411 | 0.247 | 0.170 | 0.172 |
| Router | 13.095 | 6.099 | 3.101 | 2.361 | 0.531 | 0.247 | 0.126 | 0.096 |
| Power | 15.601 | 9.307 | 7.499 | 8.495 | 0.381 | 0.228 | 0.183 | 0.208 |
| PGP | 118.751 | 94.531 | 77.171 | 53.248 | 0.346 | 0.275 | 0.225 | 0.155 |
| Email | 124.275 | 119.661 | 102.514 | 61.290 | 0.305 | 0.293 | 0.251 | 0.150 |
| Email contact | 96.511 | 95.686 | 77.846 | 12.229 | 0.342 | 0.339 | 0.276 | 0.043 |
| Hamster | 235.067 | 146.457 | 166.323 | 111.547 | 0.356 | 0.222 | 0.252 | 0.169 |
| Astro | 599.423 | 418.844 | 334.824 | 295.730 | 0.364 | 0.254 | 0.203 | 0.179 |

where $0 \leq \alpha, \delta, \mu \leq 1$. Then, we will talk about the impact of these three parameters on the node ranking as far as the spreading capability is concerned, the simulation results on 5 real networks (Astro, Email, Netsci, PGP and Power) have been displayed in Fig. 2, where each row represents all possible combinations among three kinds of neighbors and each panel exhibits the ranking results under different weight assignment. By observing all panels, we can find that $\alpha > \mu$ leads to the better ranking correlation in the first column where $\delta$ is fixed, and $\alpha > \delta$ creates the better result in the second column where $\mu$ is set to a constant. Similarly, the optimal ranking results are obtained when $\delta > \mu$ in the third column where $\alpha$ is constant. Summarizing all possible cases, $\alpha > \delta > \mu$ can result in the best ranking scenario which further demonstrate that neighbors closer to the core will contribute to the spreading capability of the present node, that is, these neighbors are vital to the diffusion of the disease.

In order to further distinguish between upper and lower neighbors with the same $k_S$ values, the spreading capability (SC) obtained in the standard SIR model is used as a reference and the average spreading capability of neighbors of each node has been computed, then these average values can be utilized as a metric to characterize the node spreading capability, which have been summarized in Table 5. In Table 5, $U_{SC}$ represents the average spreading capability of upper neighbors inside the network, which is statistically derived through the standard SIR model. Similarly, we can define $EU_{SC}$, $EL_{SC}$ and $L_{SC}$ as the average spreading capability of equal upper, equal lower and lower neighbors within the whole network. In addition, $U_{nm}$, $EU_{nm}$, $EL_{nm}$ and $L_{nm}$ denote their corresponding normalized values. From Table 5, it is shown that $U_{SC} > EU_{SC} > EL_{SC} > L_{SC}$ universally holds except the results of Hmaster data sets, that is, the upper neighbors contribute much more to the spreading capability of focal nodes than lower neighbors even if their $k_S$ is identical. Therefore, it is strongly demonstrated that the assumption about what 4 classes of neighbors contribute to the individual spreading capability is true (i.e., $\alpha > \beta > \gamma > \mu$). At the same time, too much difference between these weights will not render the optimal ranking about the individual spreading capability. Without loss of generality, we set the weight for 4 types of neighbors when we assess the individual spreading capability to be $\alpha = 0.4, \beta = 0.35, \gamma = 0.25$ and $\mu = 0.1$, that is, the estimating formula about the spreading capability can be written as

$$k_S^{CN} = 0.4 \times e^u + 0.35 \times e^{eu} + 0.25 \times e^{el} + 0.1 \times e^l \tag{4}$$

After normalizing these 4 parameters, we can rewrite Eq. (4) as

$$k_S^{CN} = 0.364 \times e^u + 0.318 \times e^{eu} + 0.227 \times e^{el} + 0.091 \times e^l \tag{5}$$

In what follows, we fix these weight assignment and compare the results obtained from Eq. (5) and those from various methods, and find that our method is a little more advantageous than degree centrality, $k$-shell and MDD algorithms.

### 3.2. Ranking similarity validation for various methods

In this subsection, we check the ranking of individual spreading capability from our method, degree centrality, $k$-shell and MDD methods, and calculate the similarity between these results and those obtained from the simulation of standard SIR epidemic model. Fig. 3 illustrates this kind of ranking similarity derived for different infection rates, where the above-mentioned 4 methods are all implemented over all 9 real-world networks. These results may display some extent of variation under different infection rate $\lambda$, but the qualitative ranking tendency is almost same for various methods. Moreover, we set $\lambda$ to be a little larger than the corresponding epidemic threshold $\lambda_C$ in all simulations for different networks since much smaller $\lambda$ leads to the extinction of outbreaks, and too large $\lambda$ renders most of nodes to be infected and the role of topology in the spreading will be overridden. As far as the similarity of ranking over the information impact range to the standard SIR model is concerned, our method is nearly all optimal over 9 real-world networks. In particular, when $\lambda > \lambda_C$, the ranking similarity produced by our method is much better than those under other methods (including degree centrality, $k$-shell and MDD), and this advantage is especially obvious in Router networks.

Here, we explore the ranking similarity by use of Kendall $\tau$ correlation coefficient [64,65]. Assume $(x_1, y_1)$, $(x_2, y_2)$, ..., $(x_n, y_n)$ denote a set of joint ranking from two ranking lists, $X$ and $Y$, respectively, and $n$ is the number of items within the data set $X$ or $Y$. If the ranking of two elements is consistent, that is, $x_i > x_j$ and $y_i > y_j$ or $x_i < x_j$ and $y_i < y_j$, it is said to be concordant for the ranking pair $(x_i, y_i)$ and $(x_j, y_j)$. Conversely, $(x_i, y_i)$ and $(x_j, y_j)$ will be discordant if $x_i > x_j$ and $y_i < y_j$

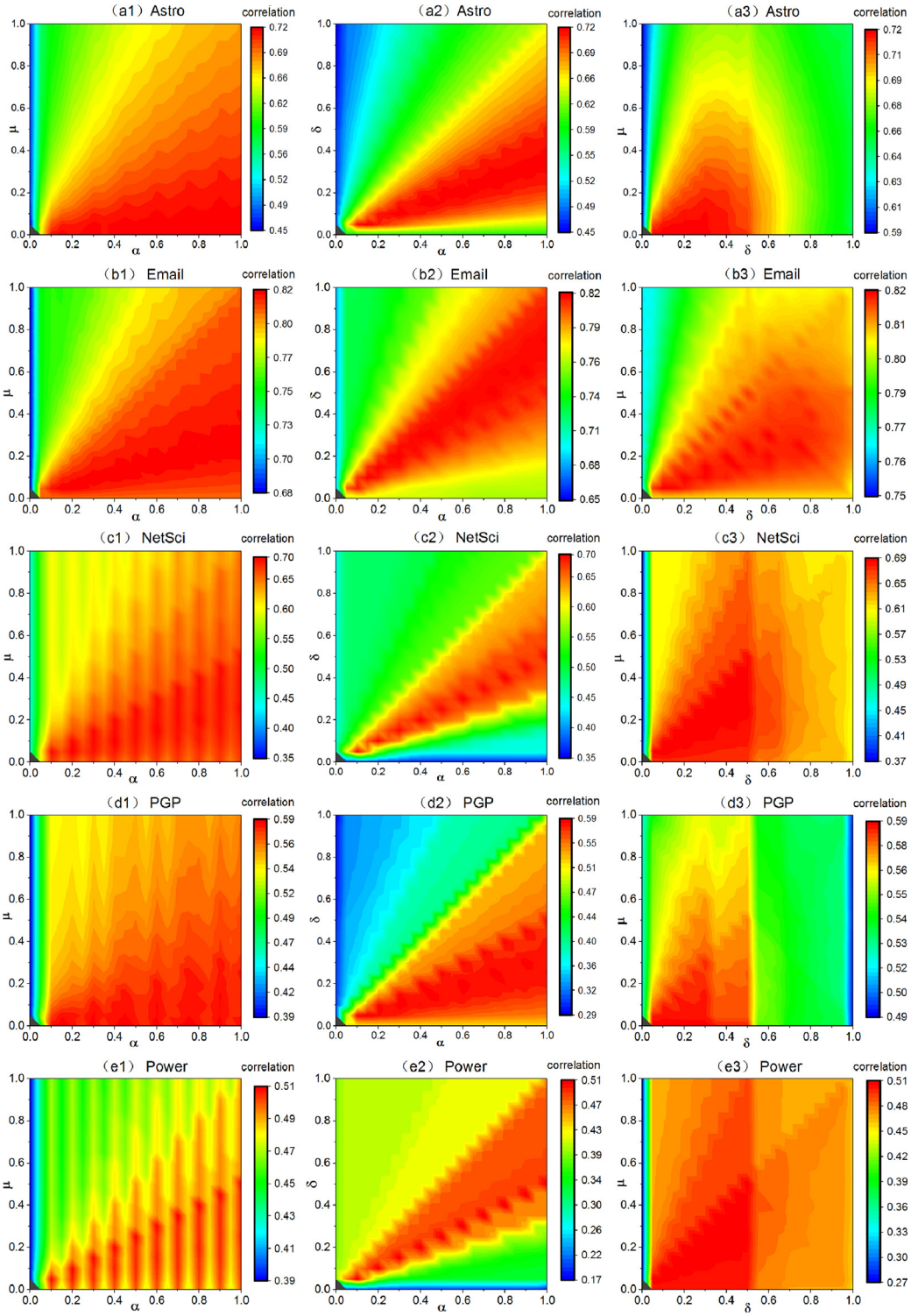


**Fig. 2.** Impact of weight distribution on the ranking similarity between our method and SIR model on 5 real-world networks as far as the spreading capability is concerned. From top to bottom, 5 real-world networks are Astro, Email, NetSci, PGP, and Power, respectively. At each row, from left to right, 2 parameters in our 3 model parameters ($\alpha$, $\mu$, $\delta$) are varied, and the color value at each point denotes the maximal correlation between two methods. In the SIR simulation, $\lambda$ is set to be a specific value shown in Table 4.

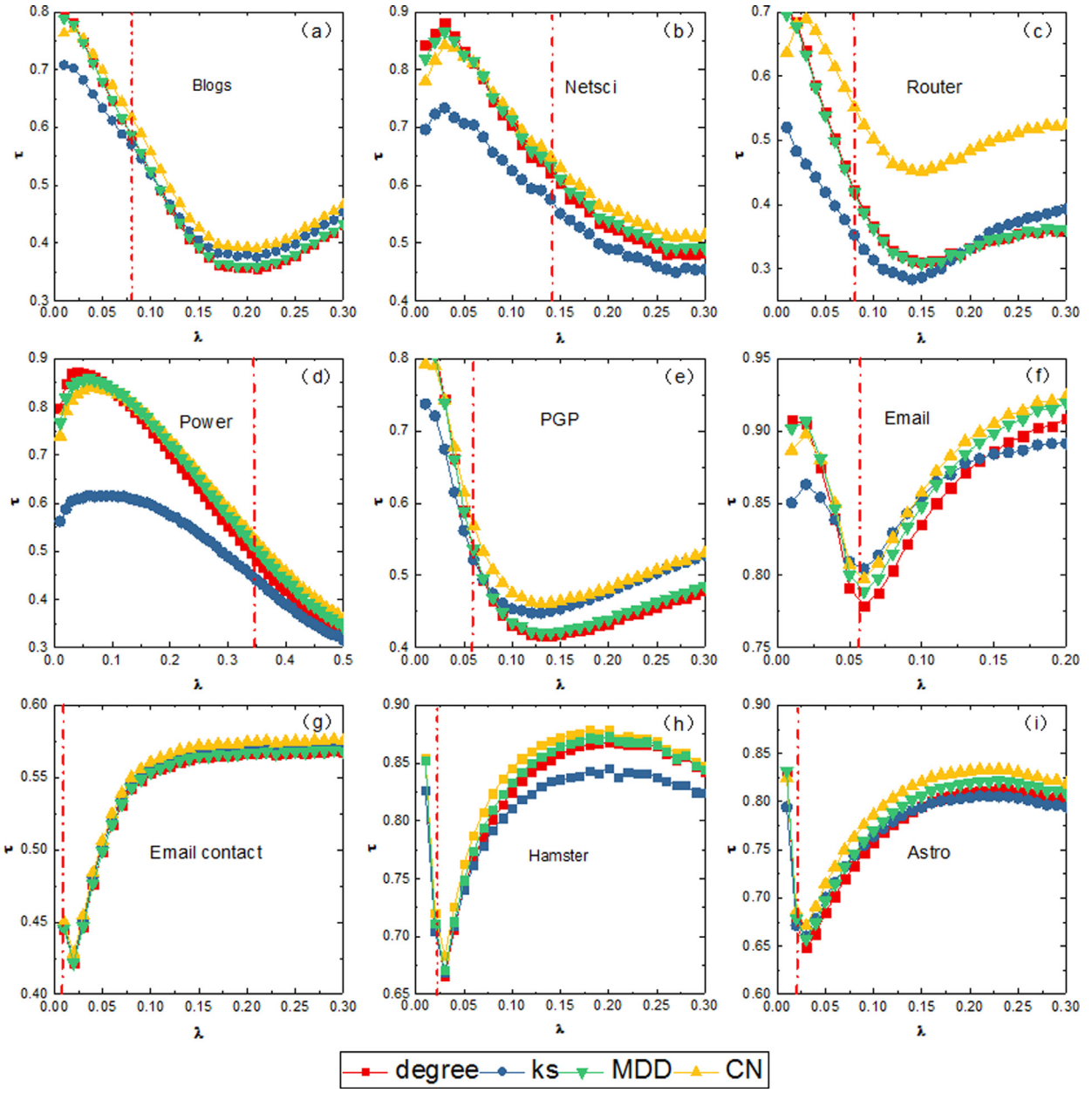


**Fig. 3.** Comparisons of ranking similarity between various methods and SIR model over 9 real-world networks. The dotted vertical line indicates the critical epidemic threshold $\lambda_C$ shown in Table 4. In addition, all parameters for our model is set according to Eq. (5). In MDD, the tunable parameter is set to be 0.7. After $\lambda > \lambda_C$, our method exhibits a relative advantage over other algorithms.

or $x_i < x_j$ and $y_i > y_j$. Moreover, if there exists the same item value within this pair (i.e., $x_i = x_j$ or $y_i = y_j$), this pair will become a tied one, which is neither concordant or discordant. Henceforth, Kendall $\tau$ coefficient can be defined by counting the number of concordant and discordant pairs between two data sets ($X$ and $Y$) as follows

$$\tau(X,Y) = \frac{n_c - n_d}{\sqrt{(n_3 - n_1)(n_3 - n_2)}} \tag{6}$$

where $n_c$ and $n_d$ denote the number of concordant and discordant pairs, respectively. $n_3 = n(n-1)/2$ represents the total number of pairs of items, $n_1 = \sum_i t_i(t_i - 1)/2$ and $n_2 = \sum_j t_j(t_j - 1)/2$, where $t_i$ is the number of tied ones in the $i$th group of ties within $X$ set and $t_j$ denotes the corresponding one within $Y$ set. Generally, $\tau$ lies in $[-1, 1]$, $\tau > 0$ characterizes the positive correlation and $\tau < 0$ means the negative correlation. For values close to 1, there exists the strong agreement

**Table 6**
The monotonicity of the ranks obtained from different algorithms over 9 real-world networks.

| Network/M(X)/Algrithm | Degree | *k*-shell | MDD | CN |
|---|---|---|---|---|
| Blogs | 0.5654 | 0.4670 | 0.5906 | 0.6876 |
| NetSci | 0.7642 | 0.6421 | 0.8226 | 0.9244 |
| Router | 0.2886 | 0.0691 | 0.3009 | 0.5777 |
| Power | 0.5927 | 0.2460 | 0.694 | 0.7716 |
| PGP | 0.6193 | 0.4806 | 0.6679 | 0.8069 |
| Email | 0.8874 | 0.8088 | 0.9233 | 0.9407 |
| Email contact | 0.1106 | 0.1090 | 0.1109 | 0.1160 |
| Hamster | 0.9247 | 0.8931 | 0.9604 | 0.9846 |
| Astro | 0.9145 | 0.8875 | 0.9435 | 0.9765 |

**Table 7**
Comparison on the running time of different algorithms over 9 real-world networks. All testing programs are implemented with Java (JDK1.8), and run on the 64-bit Windows operating systems. The system hardware includes the Intel Core(TM) *i*7 – 6700 CPU and 16 GB running memory.

| Network/Time(ms)/Algrithm | Degree | *k*-shell | MDD | CN |
|---|---|---|---|---|
| Blogs | 94 | 3422 | 24,195 | 5929 |
| Netsci | 15 | 251 | 747 | 636 |
| Router | 82 | 3366 | 23,442 | 5269 |
| Power | 81 | 3697 | 15,841 | 5386 |
| PGP | 208 | 29,120 | 161,884 | 36,547 |
| Email | 29 | 2037 | 8772 | 3003 |
| Email contact | 220 | 16,568 | 91,122 | 20,526 |
| Hamster | 34 | 5232 | 24,287 | 7137 |
| Astro | 317 | 96,446 | 719,300 | 119,126 |

between two data sets, while values close to $-1$ indicate the strong disagreement between them. Thus, $\tau$ is an index for quantifying the correspondence property of ranking similarity between two ordering lists.

### 3.3. Monotonicity validation

When we perform the ranking for the node importance according to the above methods, many nodes may hold the same ranking value, in particular for *k*-shell method in that all nodes in the same layer have the identical $k_S$ value. Thus, it is hard to differentiate among them and the complementary cumulative distribution function (CCDF) is used to examine the distribution of the ranking value to further make a distinction between similar nodes in this subsection. Generally speaking, CCDF often measures the probability in which a random variable is greater than a given value,

$$CCDF(Z) = Prob(Z > z) = 1 - CDF(z) \tag{7}$$

where $CDF(z)$ denotes the cumulative distribution function (CDF), which stands for the probability of a random variable being equal to or less than a specific value [i.e., $CDF(Z) = Prob(Z \leq z)$]. Here, starting from CCDF, we can derive the number of nodes whose ranking is higher than $z$ if $z$ is a ranking value. Furthermore, we will combine the number of ranking values if there exist some nodes which have the same ranking value. Over 9 real-world networks, CCDF regarding the similarity ranking has been summarized in Fig. 4, it can be clearly observed that the number of ranking values created by neighbor classification algorithm is much more than those produced by other three algorithms, and thus our method can more easily make a difference among the nodes within the network.

To quantitatively enhance the discriminating degree of ranking regarding the node spreading capability, we take use of the ranking monotonicity [50] to further measure the quality of different methods, and this ranking monotonicity can be calculated as follows,

$$M(X) = \left[1 - \frac{\sum_{r \in V} N_r(N_r - 1)}{N(N-1)}\right]^2 \tag{8}$$

where $X$ denotes the ranking list, $r$ is the ranking value, $N$ is the network size and $N_r$ stands for the number of nodes with the ranking value $r$. When $M(X)$ is close to 1.0, it means that this ranking list owns the better monotonicity and can well differentiate between nodes. On the contrary, if $M(X)$ approaches 0.0, it applies that there is only one ranking value within this network, that is, all nodes have the same ranking value. In Table 6, we depict the ranking monotonicity of all methods over 9 real-world networks, it can be clearly shown that our method becomes much more superior over the degree centrality, *k*-shell and MDD algorithm in terms of the ranking monotonicity.

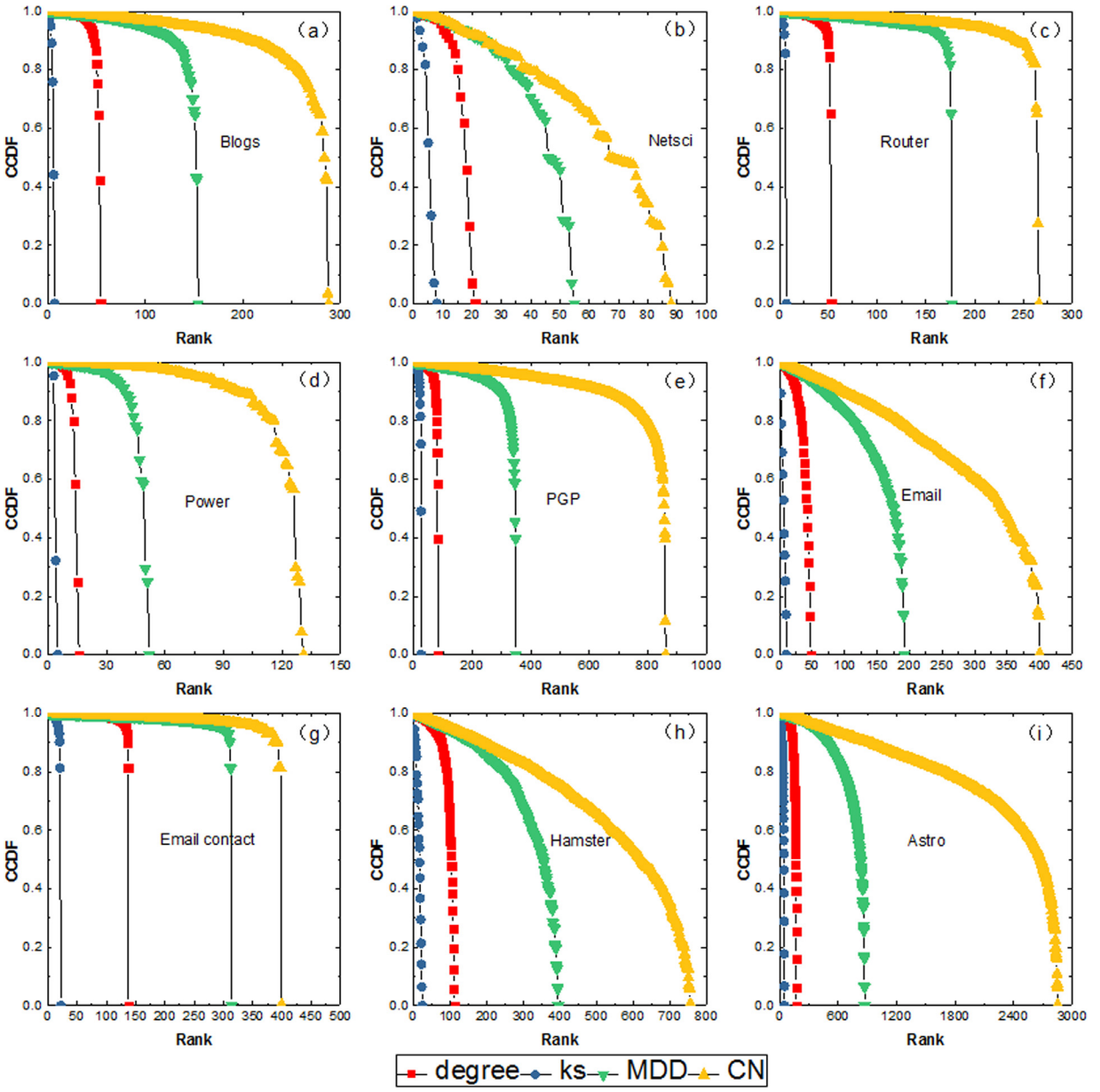


**Fig. 4.** CCDF obtained from 4 different methods evaluating the ranking similarity over 9 real-world networks. The simulation parameters are identical with those set in Fig. 3. In all networks, the simulation results indicate that our method owns a wider range of ranking numbers when compared to other three algorithms.

### *3.4. Comparison of time complexity*

In this subsection, we probe into the algorithm performance from the perspective of time complexity. Here we adopt the linked list as the storage structure of network. On the one hand, our method is based on the classical $k$-shell algorithm and we need to record the nodal removal sequence during the shell decomposition. However, the node removal order is completed during the time that we calculate the $k$-shell value, and thus the time complexity of our method is still identical with the $k$-shell algorithm $O(N+E)$ although the time constant may be higher. On the other hand, our method is superior to MDD method which needs to update its mixed degree $k_m$ at each decomposition. To deeply compare the performance of various methods, we count the real time consumption of 1000 independent runs over 9 networks and the related results are summarized in Table 7, and all tests are running on top of the same hardware and software platform. Obviously, the degree centrality is the best one regarding the time consumption but other aspects own some extent of inferiority. However, the

running time of our method on classified neighbors is a little longer than that of the classical *k*-shell, but is much shorter than that of MDD. Taking together, we obtain the higher performance with the moderate cost of time consumption when compared to the *k*-shell or MDD methods.

## 4. Conclusions

In summary, on the basis of the classical *k*-shell method, we proposed a novel identification algorithm of influential spreaders within complex networks, and large quantities of numerical simulations have been performed on 9 real-world networks and we guage the nodal spreading capability based on the standard SIR epidemic model. The results indicate that the ranking accuracy of our method is higher and the ranking differentiation is also more fine-grained when compared to other methods, such as degree centrality, *k*-shell and MDD algorithms. Additionally, the time complexity is also intermediate since the real running time data reveal that our methods often lies between *k*-shell and MDD methods, which is much lower than that of other algorithms such as closeness and betweenness centrality. The current results are much more beneficial for us to deeply understand and analyze the individual spreading capability, and even help to devise some efficient strategies to accelerate some kind of information spreading and decelerate the diffusion of ill information, or inhibit the epidemic outbreaks within the population.

## Acknowledgments

This project is partially supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61773286, 61374169 and 61403280.

## References

[1] R. Albert, A.L. Barabási, Statistical mechanics of complex networks, Rev. Mod. Phys. 74 (1) (2002) 47–97.
[2] X. Wang, G. Chen, Complex networks: small-world, scale-free and beyond, IEEE circuits and systems magazine 3 (1) (2003) 6–20.
[3] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.U. Hwang, Complex networks: structure and dynamics, Phys. Rep. 424 (4) (2006) 175–308.
[4] S.N. Dorogovtsev, A.V. Goltsev, J.F. Mendes, Critical phenomena in complex networks, Rev. Mod. Phys. 80 (4) (2008) 1275.
[5] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, C. Zhou, Synchronization in complex networks, Phys. Rep. 469 (3) (2008) 93–153.
[6] C. Castellano, S. Fortunato, V. Loreto, Statistical physics of social dynamics, Rev. Mod. Phys. 81 (2) (2009) 591.
[7] M. Perc, A. Szolnoki, Coevolutionary games: a mini review, BioSystems 99 (2) (2010) 109–125.
[8] S. Boccaletti, G. Bianconi, R. Criado, C.I. Del Genio, J. Gómez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, M. Zanin, The structure and dynamics of multilayer networks, Phys. Rep. 544 (1) (2014) 1–122.
[9] C. Wang, L. Wang, J. Wang, S. Sun, C. Xia, Inferring the reputation enhances the cooperation in the public goods game on interdependent lattices, Appl. Math. Comput. 293 (2017) 18–29.
[10] C. Xia, X. Meng, Z. Wang, Heterogeneous coupling between interdependent lattices promotes the cooperation in the prisoners dilemma game, Plos One 10 (6) (2015) e0129542.
[11] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, A. Vespignani, Epidemic processes in complex networks, Rev. Mod. Phys. 87 (3) (2015) 925.
[12] L. Wang, X. Li, Spatial epidemiology of networked metapopulation: an overview, Chin. Sci. Bull. 59 (28) (2014) 3511–3522.
[13] R. Pastor-Satorras, A. Vespignani, Epidemic spreading in scale-free networks, Phys. Rev. Lett. 86 (14) (2001) 3200–3203.
[14] J. Joo, J.L. Lebowitz, Behavior of susceptible-infected-susceptible epidemics on heterogeneous networks with saturation, Phys. Rev. E 69 (6) (2004) 066105.
[15] R. Olinky, L. Stone, Unexpected epidemic thresholds in heterogeneous networks: the role of disease transmission, Phys. Rev. E 70 (3) (2004) 030902.
[16] X. Li, X. Wang, Controlling the spreading in small-world evolving networks: stability, oscillation, and topology, IEEE Trans. Autom. Control 51 (3) (2006) 534–540.
[17] X. Fu, M. Small, D.M. Walker, H. Zhang, Epidemic dynamics on scale-free networks with piecewise linear infectivity and immunization, Phys. Rev. E 77 (3) (2008) 036113.
[18] C. Xia, L. Wang, S. Sun, J. Wang, An SIR model with infection delay and propagation vector in complex networks, Nonlinear Dyn. 69 (3) (2012) 927–934.
[19] C. Xia, Z. Wang, J. Sanz, S. Meloni, Y. Moreno, Effects of delayed recovery and nonuniform transmission on the spreading of diseases in complex networks, Phys. A Stat. Mech. Appl. 392 (7) (2013) 1577–1585.
[20] M. Boguná, R. Pastor-Satorras, A. Vespignani, Absence of epidemic threshold in scale-free networks with degree correlations, Phys. Rev. Lett. 90 (2) (2003) 028701.
[21] L. Yang, X. Yang, J. Liu, Q. Zhu, C. Gan, Epidemics of computer viruses: a complex-network approach, Appl. Math. Comput. 219 (16) (2013) 8705–8717.
[22] Q. Guo, X. Jiang, Y. Lei, M. Li, Y. Ma, Z. Zheng, Two-stage effects of awareness cascade on epidemic spreading in multiplex networks, Phys. Rev. E 91 (1) (2015) 012822.
[23] Q. Guo, Y. Lei, C. Xia, L. Guo, X. Jiang, Z. Zheng, The role of node heterogeneity in the coupled spreading of epidemics and awareness, Plos One 11 (8) (2016) e0161037.
[24] R. Pastor-Satorras, A. Vespignani, Immunization of complex networks, Phys. Rev. E 65 (3) (2002) 036104.
[25] E. Ahmed, A. Hegazi, A. Elgazzar, An epidemic model on small-world networks and ring vaccination, Int. J. Mod. Phys. C 13 (02) (2002) 189–198.
[26] R. Cohen, S. Havlin, D. Ben-Avraham, Efficient immunization strategies for computer networks and populations, Phys. Rev. Lett. 91 (24) (2003) 247901.
[27] D. Zhao, L. Wang, S. Li, Z. Wang, L. Wang, B. Gao, Immunization of epidemics in multiplex networks, PloS One 9 (11) (2014) e112018.
[28] Z. Wang, C. Bauch, S. Bhattacharyya, A. d'Onofrio, P. Manfredi, M. Perc, N. Perra, Ma. Salathé, D. Zhao, Statistical physics of vaccination, Phys. Rep. 664 (2016) 1–113.
[29] P. Wang, A.L. Barabási, Understanding the spreading patterns of mobile phone viruses, Science 324 (2009) 1071–1076.
[30] H. Hu, S. Myers, V. Colizza, A. Vespignani, Wifi networks and malware epidemiology, Proc. Natl. Acad. Sci. 106 (5) (2009) 1318–1323.
[31] A. Garas, P. Argyrakis, C. Rozenblat, M. Tomassini, S. Havlin, Worldwide spreading of economic crisis, New J. Phys. 12 (11) (2010) 113043.
[32] R. Baldick, B. Chowdhury, I. Dobson, et al., Vulnerability assessment for cascading failures in electric power systems, in: Proceedings of the Power Systems Conference and Exposition, PSCE'09, IEEE, 2009, pp. 1–9. IEEE/PES.
[33] J. Kostka, Y. Oswald, R. Wattenhofer, Word of mouth: rumor dissemination in social networks, Int. Colloq. Struct. Inf. Commun. Complex. 46 (1) (2008) 185–196.
[34] H. Ma, Y. Zhu, D. Li, D. Kim, J. Liang, Improving the influence under ic-n model in social networks, Discr. Math. Algorit. Appl. 7 (03) (2015) 1550037.
[35] P. Kumaran, S. Chitrakala, A survey on influence spreader identification in online social network, in: Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES), IEEE, 2016, pp. 1–7.

[36] H. Wang, F. Wang, K. Xu, Modeling information diffusion in online social networks with partial differential equations, Comput. Sci. 42 (2) (2013) 31–36.
[37] D. Kempe, J. Kleinberg, 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, ACM, 2003, pp. 137–146.
[38] S. Hong, H. Yang, T. Zhao, X. Ma, Epidemic spreading model of complex dynamical network with the heterogeneity of nodes, Int. J. Syst. Sci. 47 (11) (2016) 2745–2752.
[39] W. Wang, M. Tang, H. Zhang, H. Gao, Y. Do, Z. Liu, Epidemic spreading on complex networks with general degree and weight distributions, Phys. Rev. E 90 (4) (2014) 042803.
[40] Z. Shen, S. Cao, W. Wang, Z. Di, H.E. Stanley, Locating the source of diffusion in complex networks by time-reversal backward spreading, Phys. Rev. E 93 (3) (2016) 032301.
[41] P. Bonacich, Factoring and weighting approaches to status scores and clique identification, J. Math. Sociol. 2 (1) (1972) 113–120.
[42] L.C. Freeman, Centrality in social networks conceptual clarification, Soc. Netw. 1 (3) (1978) 215–239.
[43] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry 40 (1) (1977) 35–41.
[44] L. Katz, A new status index derived from sociometric analysis, Psychometrika 18 (1) (1953) 39–43.
[45] B. Bollobás, Graph Theory and Combinatorics: Proceedings of the Cambridge Combinatorial Conference in Honour of Paul Erdös, [Trinity College, Cambridge, 21–25 March 1983], Academic Press, 1984.
[46] S.B. Seidman, Network structure and minimum degree, Soc. Netw. 5 (3) (1983) 269–287.
[47] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, E. Shir, A model of internet topology using *k*-shell decomposition, Proc. Natl. Acad. Sci. 104 (27) (2007) 11150–11154.
[48] M. Kitsak, L.K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H.E. Stanley, H.A. Makse, Identification of influential spreaders in complex networks, Nat. Phys. 6 (11) (2010) 888.
[49] L. Zhong, C. Gao, Z. Zhang, N. Shi, J. Huang, Identifying influential nodes in complex networks: a multiple attributes fusion method, in: Proceedings of the International Conference on Active Media Technology, Springer, 2014, pp. 11–22.
[50] J. Bae, S. Kim, Identifying and ranking influential spreaders in complex networks by neighborhood coreness, Phys. A Stat. Mech. Appl. 395 (2014) 549–559.
[51] L. Ma, C. Ma, H. Zhang, B. Wang, Identifying influential spreaders in complex networks based on gravity formula, Phys. A Stat. Mech. Appl. 451 (2016) 205–212.
[52] A. Zeng, C. Zhang, Ranking spreaders by decomposing complex networks, Phys. Lett. A 377 (14) (2013) 1031–1035.
[53] Y. Liu, M. Tang, T. Zhou, Y. Do, Core-like groups result in invalidation of identifying super-spreader by *k*-shell decomposition, Sci. Rep. 5 (5) (2015) 9602.
[54] Y. Liu, M. Tang, T. Zhou, Y. Do, Improving the accuracy of the *k*-shell method by removing redundant links: from a perspective of spreading dynamics, Sci. Rep. 5 (2015) 13172.
[55] Y. Moreno, R. Pastor-Satorras, A. Vespignani, Epidemic outbreaks in complex heterogeneous networks, Eur. Phys. J. B-Condens. Matter Complex Syst. 26 (4) (2002) 521–529.
[56] B. Macdonald, P. Shakarian, N. Howard, G. Moores, Spreaders in the network SIR model: an empirical study, arXiv preprint arXiv:1208.4269(2012).
[57] J. Duch, A. Arenas, Community detection in complex networks using extremal optimization, Phys. Rev. E 72 (2) (2005) 027104.
[58] M.E. Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74 (3) (2006) 036104.
[59] N. Spring, R. Mahajan, D. Wetherall, Measuring ISP topologies with rocketfuel, ACM SIGCOMM Comput. Commun. Rev. 32 (4) (2002) 133–145.
[60] D.J. Watts, S.H. Strogatz, Collective dynamics of'small-world'networks, Nature 393 (6684) (1998) 440.
[61] M. Boguñá, R. Pastor-Satorras, A. Díaz-Guilera, A. Arenas, Models of social networks based on social distance attachment, Phys. Rev. E 70 (5) (2004) 056122.
[62] R. Guimera, L. Danon, A. Diaz-Guilera, F. Giralt, A. Arenas, Self-similar community structure in a network of human interactions, Phys. Rev. E 68 (6) (2003) 065103.
[63] M.E.J. Newman, The structure of scientific collaboration networks, Proc. Natl. Acad. Sci. 98 (2) (2001) 404–409.
[64] M.G. Kendall, A new measure of rank correlation, Biometrika 30 (1/2) (1938) 81–93.
[65] M.G. Kendall, The treatment of ties in ranking problems, Biometrika 33 (3) (1945) 239–251.