# Optimization of targeted node set in complex networks under percolation and selection

Yang Liu,[1,2,*] Xi Wang,[3] and Jürgen Kurths[1,4,5]

[1]*Potsdam Institute for Climate Impact Research, 14412 Potsdam, Germany*

[2]*Department of Computer Science, Technische Universität Berlin, 10587 Berlin, Germany*

[3]*Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong*

[4]*Department of Physics, Humboldt University Berlin, 12489 Berlin, Germany*

[5]*Institute of Applied Physics, Russian Academy of Science, 603950 Nizhny Novgorod, Russia*

(Received 22 February 2018; revised manuscript received 1 June 2018; published 23 July 2018)

Most of the existing methods for the robustness and targeted immunization problems can be viewed as greedy strategies, which are quite efficient but readily induce a local optimization. In this paper, starting from a percolation perspective, we develop two strategies, the relationship-related (RR) strategy and the prediction relationship (PR) strategy, to avoid a local optimum only through the investigation of interrelationships among nodes. Meanwhile, RR combines the sum rule and the product rule from explosive percolation, and PR holds the assumption that nodes with high degree are usually more important than those with low degree. In this manner our methods have a better capability to collapse or protect a network. The simulations performed on a number of networks also demonstrate their effectiveness, especially on large real-world networks where RR fragments each of them into the same size of the giant component; however, RR needs only less than 90% of the number of nodes which are necessary for the most excellent existing methods.

DOI: 10.1103/PhysRevE.98.012313

## I. INTRODUCTION

There has recently been an enormous amount of interest focusing on the targeted immunization and robustness problems of network science [1–5], like investigating the critical threshold of structural collapse if an intentional attack happens [6], or probing the optimal targeted-immunized threshold if a virus is in possible transmission [7]. These problems appear in, but are not limited to, effectively preventing viruses in computer or population related networks [8,9], information transmission in social networks [10–12], or the breakdown of some infrastructure networks [13,14].

For a network, the solution to the critical or optimal threshold is mathematically equivalent to finding the minimum set of nodes which can fragment the network into a certain situation, e.g., the size of the giant component is less than a given value after the removal of the minimum set. To achieve this, numerous methods have been proposed in the last few years, consisting of random immunization [15], acquaintance strategies [7,16], targeted methods [3,4,17–19], etc. [20–23], ranging from the need of local information to the whole network demand. With respect to random immunization, the immunization nodes are randomly selected from a certain network—without any priority about them. Similarly, random selection is also applied in the acquaintance strategy, but only one of the neighbors of a certain node is chosen to be immunized [7]. In addition, the targeted method is a widely accepted approach which first identifies the importance of each node and then removes the nodes in descending order of importance until the network reaches the immunized demand [3,4,18,19].

Within networks, there are numerous relationships among nodes. Generally, high-degree nodes tend to connect to other high-degree nodes in assortatively mixed networks, while they mostly have low-degree neighbors in disassortative networks [24]. Moreover, a node with a low degree might play a critical role, whereas those with high degree might not be of significance comparatively (e.g., the betweenness centrality of nodes [25]). In the course of immunization, some subinfluential nodes would become influential after a few nodes are removed, while some others might lose their importance instead [heuristic immunized strategies [3,4,18], including the high adaptive degree centrality strategy, etc.]. All of such methods, e.g., the Collective Influence method (CI) [3] (better results always obtained with larger radius $\ell$), show that a better immunization strategy could be discovered when more interrelationships of nodes are considered. This may be also a good interpretation why the high adaptive betweenness centrality strategy (HAB) is significantly effective in most situations, as well as the belief propagation-guided decimation (BPD) method [4,26] in artificial networks. But HAB has a limitation due to its high time complexity [$O(n^2m)$] and BPD is not so effective in real-world networks because there are always many loops.

Most of those methods can also be viewed as greedy strategies, i.e., they repeat the process that recalculates the importance of nodes in the remaining network and then remove the most influential one or a part of it. For an optimization problem, the greedy strategy is quite efficient but readily induces a local optimum. In addition, taking Fig. 1(a) as an example, the removal of a node would affect the status (remove or not) of other nodes. These facts motivate us to use another approach: can the local optimization be effec-

*yangliu@pik-potsdam.de

 

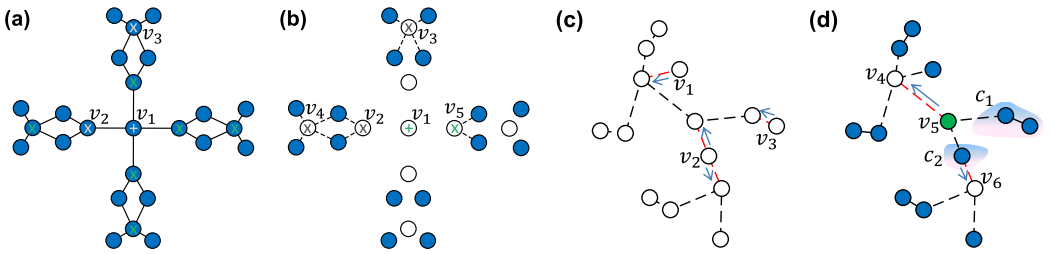


FIG. 1. Brief illustrations of the proposed methods. (a) In this network almost all of the methods mentioned in this paper will remove $v_1$ (marked with "+") in the demand case of splitting this network into isolated ones, while the optimal removal set should be the nodes marked by "×" apparently. Now, assuming that $v_2$ is removed first, then most of these methods will easily find the optimal solution in the remaining network. But the same situation cannot be induced by removing $v_3$. In other words, the removal of $v_2$ or $v_3$ will influence the status of $v_1$ (remove or not), and as a result it directly determines whether the optimal solution can be reached. (b) An example of RR under the sum rule and $\tau = 2$. In this temporary network ($\kappa = 16$), we consider two assumed cases: (1) $(v_3, v_4)$ then $(v_1, v_5)$ and (2) $(v_2, v_4)$, then $(v_1, v_5)$, i.e., two rounds of selection. For the first choice between $v_3$ and $v_4$ in case 1, the occupied node is either $v_3$ or $v_4$ since $\xi(v_3) = \xi(v_4) = 5$. After this, node $v_1$ would be chosen because of $1 = \xi(v_1) < \xi(v_5) = 3$ (might induce the optimal solution of $q_c$). In contrast, $v_2$ would be selected to be occupied at first $[3 = \xi(v_2) < \xi(v_4) = 5]$, and then $v_5$ $[4 = \xi(v_1) > \xi(v_5) = 3]$ in case 2 (might be associated with the optimization of $F$). In this example, we can also find how the status of a node influences the status of other nodes, e.g., $v_2$ to $v_1$ and $v_5$. (c, d) An example of the PR method. In (c), some low-degree nodes are chosen and occupied first. (d) $\xi(v_5)$= 4/14 and here $\mathcal{C}_{v_5} = \{c_1, c_2\}$ marked by color shadow.

tively avoided by investigating the interrelationship among nodes?

Here, also from a percolation perspective [3,19,27,28], we propose two strategies: the relationship-related (RR) method and the prediction relationship (PR) method, which are capable of achieving excellent performance compared to other existing strategies. The main idea of the developed strategies is to explore and utilize the interrelationship among the nodes. In addition, RR combines the sum rule and the product rule from explosive percolation [29], while PR holds the assumption that nodes with high degree are usually more important than those with low degree. In this way, our approaches can achieve a better capability of avoiding of local optimum and obtain smaller thresholds than other methods. To demonstrate the effectiveness of the proposed strategies, we conduct numerous simulations on a number of networks. The results show that our methods have significant advantages over other strategies, especially on large real-world networks where RR can collapse each of them into the same size of the giant component with less than 90% nodes of CI, BPD, or the Explosive Immunization method (EI) [19]. Moreover, our methods might also be used for the feedback vertex set (FVS) problem [26,30,31].

## II. METHOD

We consider an undirected network composed of $n = |\mathcal{N}|$ nodes tied by $m = |\mathcal{M}|$ edges where $\mathcal{N}$ and $\mathcal{M}$ are the node set and the edge set, accordingly. Let $S_a$ be an arbitrary configuration (sequence) of $\mathcal{N}$, namely, $\{S_a(i), i \in [1,n]\} \equiv \mathcal{N}$ where $S_a(i)$ corresponds to a unique node of the network. Then the threshold $q_c^{S_a}$ regarding $S_a$ is defined to be

$$q_c^{S_a} := \min_q \{q \in [0,1] | G(S_a; q) \leqslant \epsilon\}, \tag{1}$$

in which $\epsilon$ is a given value and $G(S_a; q)$ represents the probability that a node is part of the giant (largest) connected component in the remaining network after the removal of all nodes in $\{S_a(i), i \in [1, \lfloor n \times q \rfloor]\}$, including the incidental edges. Denoting by $F^{S_a}$,

$$F^{S_a} := \frac{1}{n} \sum_q^1 G(S_a; q), \tag{2}$$

the average size fraction of giant components of $S_a$, the solution associated with the targeted immunization or robustness problem is to search the optimal sequence $S_\theta$, which satisfies

$$S_\theta \equiv \begin{cases} \{S_i | q_c^{S_i} \leqslant q_c^{S_a}, \forall i, a\}, & \text{if } \epsilon \text{ is given,} \\ \{S_i | F^{S_i} \leqslant F^{S_a}, \forall i, a\}, & \text{otherwise,} \end{cases} \tag{3}$$

where $i, a \in [1, n!]$ mean all the configurations of $\mathcal{N}$. Apparently, finding the optimal solution is NP-hard.

### A. Relationship-related (RR) method

Inspired by Ref. [29], we develop RR method in a percolation process, i.e., change the process from the removal of the most influential node to the occupation of the least important node. In other words, we start the RR method with an arbitrary configuration $S_{a,0}$ of the node set $\mathcal{N}$ and a nonoccupied network [or a given strategy, e.g., high-degree centrality strategy (HD)], and then reverse the order of $S_{a,0}$ to be a new sequence $S'_{a,0}$, satisfying

$$S'_{a,0}(i) \equiv S_{a,0}(j), \tag{4}$$

where $i + j = n + 1, \forall i, j \in [1,n]$. Let $r$ be the proportion of possible candidates and $\tau \in \mathbb{Z}^+$ be the selection times, respectively. Denoting the number of occupied nodes with $\kappa$, we then obtain $S_{a,1}$ based on $S'_{a,0}$ through the following procedures:

(i) Each time randomly select one node $v_i$ from the nearest nonoccupied node set $\mathcal{N}^u(\kappa)$:

$$\mathcal{N}^u(\kappa) = \{S'_{a,0}(j) | j \in [\kappa + 1, \min(\lfloor \kappa + r \times n \rfloor, n)]\}. \tag{5}$$

(ii) Independently repeat the selection (i) $\tau$ times to form the candidate node set $\mathcal{N}^c(\kappa)$, and then choose the node $v_c$ from $\mathcal{N}^c(\kappa)$ which minimizes $\xi(\cdot)$ to be occupied (randomly

choose one if there are several nodes with the same minimum):

$$v_c = \arg\min_{v_j} \xi(v_j), v_j \in \mathcal{N}^c(\kappa), \tag{6}$$

where $\xi(v_j)$ is defined as the following two cases (respectively correspond to the sum rule and the product rule [29]):

$$\xi(v_j) = \begin{cases} 1 + \sum\limits_{c_i \in \mathcal{C}_{v_j}} \mathcal{G}_{c_i}, \\ 1 + \prod\limits_{c_i \in \mathcal{C}_{v_j}} \mathcal{G}_{c_i}, \end{cases} \tag{7}$$

in which $\mathcal{G}_{c_i}$ is the size of the component $c_i$ and $\mathcal{C}_{v_j}$ denotes the component set that node $v_j$ would connect in the temporary network consisting of all the occupied nodes $\{S^{\epsilon}_{a,0}(j)|j \in [1,\kappa]\}$ and the related edges.

(iii) Update $S'_{a,0}$ by swapping $S'_{a,0}(\kappa+1)$ and $v_c$, i.e., exchange the places of $S'_{a,0}(\kappa+1)$ and $v_c$ in $S'_{a,0}$.

(iv) Repeat the processes (i)–(iii) until all nodes are occupied, and we will get a new sequence $S_{a,1}$ by reversing $S'_{a,0}$.

Next, replace $S_{a,1}$ with $S_{a,0}$ based on Eq. (3), namely, replace $S_{a,1}$ with $S_{a,0}$ if $\epsilon$ is given and $q_c^{S_{a,1}} > q_c^{S_{a,0}}$, otherwise, replace $S_{a,1}$ with $S_{a,0}$ if $F^{S_{a,1}} > F^{S_{a,0}}$. In this manner, we further obtain $S_{a,2}$ based on $S_{a,1}$ as well as $S_{a,T}$ with other $r$ and $\tau$ where $T$ denotes the time step. An illustration of RR is shown in Fig. 1(b).

Now let us focus our attention on the parameters $r$ and $\tau$ as well as the two kinds of selection strategies [Eq. (7)]. A large $r$ indicates that the selection happens on a large range of possible candidates, which on the one hand can make RR converge quickly at the early stage, but on the other hand it will induce RR saturating and no longer improving after $T$ reaches some value since $\tau \ll r \times n$. The value of $\tau$ is the main contribution to the time consumption of RR. To overcome this, here we associate the $r$ and $\tau$ with $T$ (may have other choices):

$$\begin{aligned} r &= \frac{r_s}{T\delta_r + 1}, \\ \tau &= \tau_s + \lfloor T\delta_\tau + 0.5 \rfloor, \end{aligned} \tag{8}$$

where $r_s$ and $\tau_s$ are the initial values of $r$ and $\tau$, $\delta_r$ is the decrease rate of $r$ and $\delta_\tau$ denotes the increases rate of $\tau$, respectively.

With respect to the two kinds of selection strategies, our simulation results demonstrate that the sum rule is more efficient than the product rule in small networks but less efficient in large network. Hence, we combine them and use the following adaptive probability to determine which one is adopted in each $T$:

$$p_{sr} = \frac{\pi_{sr}}{\pi_{sr} + \pi_{pr}}, \tag{9}$$

where $p_{sr}$ is the selection probability of the sum rule, otherwise the product rule. $\pi_{sr}$ and $\pi_{pr}$ correspond to the number of positive replacements under the sum rule and the product rule, respectively. In other words, if the sum rule promotes a better result (smaller $F$ or $q_c$), then $\pi_{sr} = \pi_{sr} + 1$, vice versa. In this paper, we initialize $\pi_{sr}$ and $\pi_{pr}$ with 1.

### B. Prediction relationship (PR) method

The PR method is developed based on an assumption that high-degree nodes are normally more influential than those nodes with low degree, i.e., PR tries to keep the occupied components away from as many high-degree nodes as possible. To achieve this, we first identify each node based on the distribution of node degree:

$$\mathcal{H}_{v_i} = 1 - \sum_{k_{v_j} < k_{v_i}} p(k_{v_j}) = \sum_{k_{v_j} \geqslant k_{v_i}} p(k_{v_j}), \tag{10}$$

where $p(k_{v_j})$ is the probability of nodes with degree $k_{v_j}$. Then, similar to RR, construct the $\xi(\cdot)$ function with

$$\xi(v_i) = \sum_{c_i \in \mathcal{C}_{v_i}} \sum_{v_j \in c_i} \sum_{v_z \in \Gamma^u(v_j)} \mathcal{H}_{v_z} + \sum_{v_z \in \Gamma^u(v_i)} \mathcal{H}_{v_z} \tag{11}$$

in which $\Gamma^u(v_j)$ denotes all of the $v_j$'s nearest-unoccupied neighbors (here view $v_i$ as occupied node). An example of PR is shown in Figs. 1(c) and 1(d).

### C. RR and PR for the feedback vertex set (FVS) problem

Following Ref. [4], we further develop RR and PR to obtain the optimal FVS of a given network, which can help RR and PR to obtain better $q_c$ than the direct calculation in model networks. Let FVS be a subset of $\mathcal{N}$, after the removal of it there is no loop in the remaining network ($\mathcal{N} \setminus$ FVS). Denoting with $n_{\mathrm{FVS}}$ the number of nodes in FVS, the goal of optimizing FVS is to minimize $n_{\mathrm{FVS}}$. How can we achieve this in RR and PR?

Considering the candidate node set $\mathcal{N}^c(\kappa)$ [see Sec. II A (ii)], we construct the subset $\mathcal{N}^c_{\mathrm{FVS}}(\kappa)$ of it in the following way:

$$\mathcal{N}^c_{\mathrm{FVS}}(\kappa) = \{v_j | \arg\min_{v_j} \psi(v_j), v_j \in \mathcal{N}^c(\kappa)\}, \tag{12}$$

where $\psi(v_j)$ is defined as

$$\psi(v_j) = \sum_{c_i \in \mathcal{C}_{v_j}} (|\Gamma^o(v_j, c_i)| - 1) \tag{13}$$

in which $|\Gamma^o(v_j, c_i)|$ is the number of occupied nodes that belong to the component $c_i$ as well as the nearest neighbors of $v_j$. Then we rewrite Eq. (6) as

$$v_c = \arg\min_{v_j} \xi(v_j), v_j \in \mathcal{N}^c_{\mathrm{FVS}}(\kappa), \tag{14}$$

where $v_c$ corresponds to the node chosen to be occupied. In addition, another strategy is adopted for the FVS problem: if $\psi(v_c) = 1$ (this means that there are two neighbors of $v_c$ in the same component, i.e., the selected occupied node $v_c$ will induce a loop), then we further exchange the places of $v_c$ and one of its two corresponding neighbors (randomly) after the swap process [see Sec. II A (iii)]. Finally, without a loss of generality, we replace $S_{a,1}$ with $S_{a,0}$ if $S_{a,0}$ has smaller $n_{\mathrm{FVS}}$ than $S_{a,1}$.

Obviously, there is no loop in the temporary network (composing of $\{S'_{a,0}(j)|j \in [1,\kappa]\}$) if all occupied nodes satisfy $\psi(v_c) = 0$ [Eq. (14)] in the occupied process. In other words, the minimization of $n_{\mathrm{FVS}}$ is equivalent to the maximization of $\kappa$ under the constraint of $\psi(v_c) = 0$.

## III. RESULTS

In this section, if there is no special explanation, $\ell$ of CI [3] is fixed to 4, each result of EI [19] is obtained with $K = 6$

and 2000 candidates, and BPD [4] is conducted with $x = 12$. Note that the results of BPD are slightly different from the results in Ref. [4], since we fix the "Degree threshold" with the "Degree of top percent" in the BPD code (Table II). To validate the effectiveness of the proposed methods in more detail, here we test RR and PR by considering different optimization objectives, $F$ and $q_c$, respectively. In addition, both RR and PR are based on HD with $r_s = F^{\mathrm{HD}}$, $\tau_s = 10$, $\delta_r = 0.001$ and $\delta_\tau = 0.01$ for networks with $n \leqslant 10^4$, $\delta_r = 0.01$, and $\delta_\tau = 0.01$ for networks with $10^4 < n \leqslant 10^5$, $\delta_r = 0.1$, and $\delta_\tau = 0.1$ for networks with $10^5 < n \leqslant 10^6$, and $\delta_r = 0.5$ and $\delta_\tau = 0.5$ for networks with $n > 10^6$, accordingly. The threshold $q_c$ is assumed to be obtained with $G(S;q) < 0.01$.

We first conduct our validation on a number of real-world networks from various fields: one Power Grid network [32,33] (Power), three Collaboration networks [34] (including ca-GrQc, ca-AstroPh, and ca-CondMat), one Internet peer-to-peer network [34,35], Autonomous systems graphs [36] (including as-733 and as-Skitter), the Scottish cattle movements network [19], two Citation networks [36,37] (including hep-th and cit-HepTh), two Communication networks (including email-Enron [38,39] and email-EuAll [34]), one Location-based online social network [40] (loc-Gowalla), the Amazon product co-purchasing network [41] (com-Amazon), the Google web graph [39] (web-Google), and two Road networks [39] (including roadNet-PA and roadNet-TX). The choices of these networks consider both the density of edges [1] and the assortativity of degrees [24,42], which are associated with robustness of a network. Some basic information regarding these networks is given in Table I. Note that for all networks studied here, the directed edges are simply replaced with undirected edges, and self-loops and isolated nodes are entirely deleted.

In Fig. 2 the proportion $G(q)$ of the largest component versus the fraction $q$ of removed nodes is plotted by comparing RR, CI, BPD, and EI on the CA-AstroPh network, the Cit-HepPh network, the TXroad network, and the as-Skitter network. In almost all the situations studied here, RR exhibits notable superiority of less nodes to be removed for same size of giant component compared to the other strategies. Further regarding certain metrics (Fig. 3 and Table II), RR also shows better threshold $q_c$ in most networks and represents minimal average giant fraction $F$ in all cases compared to HD, CI, BPD, and EI, especially for the four largest networks where both $F$ and $q_c$ of RR are significantly smaller than the other strategies, e.g., RR needs less than half of nodes of HD, CI, and BPD to split the two road networks into fragments with $G(S;q) < 0.01$. In addition, PR can also achieve smaller $q_c$ in 12/17 networks than HD, CI, BPD, and EI.

TABLE I. Basic information of the real-world networks where CC is the clustering coefficient [32] and AC denotes the assortativity coefficient [24], respectively.

| Networks[a] | $n$ | $m$ | CC | AC |
|---|---|---|---|---|
| Power | 4941 | 6594 | 0.0801 | 0.0035 |
| CA-GrQc | 5242 | 14 484 | 0.5296 | 0.6593 |
| p2p-Gnutella08 | 6301 | 20 777 | 0.0109 | 0.0356 |
| as-733 | 6474 | 12 572 | 0.2522 | − 0.1818 |
| Scottish | 7228 | 24 784 | 0.2798 | − 0.1985 |
| CA-AstroPh | 18 771 | 198 050 | 0.6306 | 0.2051 |
| CA-CondMat | 23 133 | 93 439 | 0.6334 | 0.1340 |
| hep-th | 27 240 | 341 923 | 0.3119 | − 0.0302 |
| Cit-HepPh | 34 546 | 420 877 | 0.2848 | − 0.0063 |
| Email-Enron | 36 692 | 183 831 | 0.4970 | − 0.1108 |
| loc-Gowalla | 196 591 | 950 327 | 0.2367 | − 0.0293 |
| Email-EuAll | 265 214 | 364 481 | 0.0671 | − 0.1781 |
| com-Amazon | 334 863 | 925 872 | 0.3967 | − 0.0588 |
| web-Google | 875 713 | 4 322 051 | 0.5143 | − 0.0551 |
| PAroad | 1 088 092 | 1 541 898 | 0.0465 | 0.1227 |
| Txroad | 1 379 917 | 1 921 660 | 0.0470 | 0.1304 |
| as-Skitter | 1 696 415 | 11 095 298 | 0.2581 | − 0.0814 |

[a]The source data of these networks is from either http://www.snap.stanford.edu/data or http://www.konect.uni-koblenz.de/networks/opsahl-powergrid.

We further evaluate the performance of the proposed strategies (both RR and PR) by focusing on artificial model networks [including Erdős-Rényi (ER) [45] and scale-free (SF) [33] networks]. Note that here RR is in the normal way (optimizing $F$) and PR is to optimize FVS (following the idea of BPD). As illustrated in Fig. 4, RR significantly outperforms CI of lower $G(q)$ curves on both ER and SF networks. Considering the threshold $q_c$ on the ER networks, we respectively obtain $q_c = 0.1767$ by BPD, $q_c = 0.1843$ through EI (slightly larger, 0.0005, than the results in Ref. [19]) and $q_c = 0.1809$ with PR. Meanwhile, PR with $q_c = 0.0977$ is closer to BPD with $q_c = 0.0965$ compared to EI with $q_c = 0.0996$ in the SF networks. Besides, the results of $q_c$ versus the average degree $\langle k \rangle$ are exhibited in Fig. 5. Interestingly, when tied by $K = 6$, EI performs worse and worse with the increase of $\langle k \rangle$. The

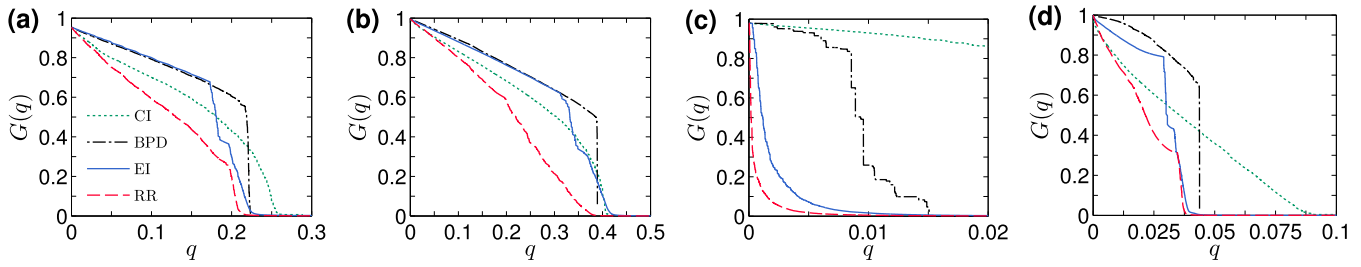


FIG. 2. The fraction $G(q)$ of the size of the giant component versus the fraction of removed nodes $q$ for CI, BPD, EI, and RR for (a) the CA-AstroPh network, (b) the Cit-HepPh network, (c) the TXroad network, and (d) the as-Skitter network (where CI with $\ell = 2$). Each result of EI and RR is obtained by averaging 20 realizations.

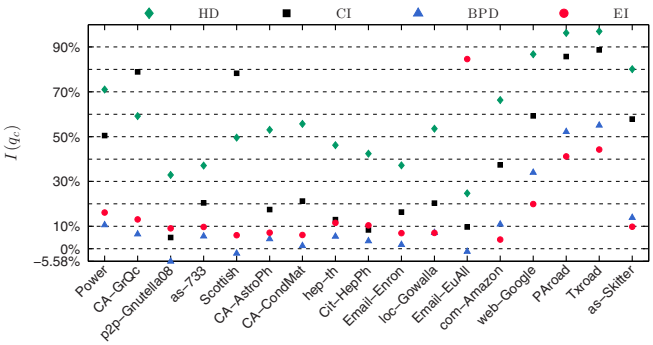


FIG. 3. The percentage of improvement $I(q_c)$ of 17 real-world networks by comparing RR with HD, CI, BPD, and EI. $I(q_c) = (q_c^{S_a} - q_c^{\mathrm{RR}})/q_c^{S_a}$ where $S_a$ corresponds to HD, CI, BPD, or EI. Each result of EI and RR is obtained by averaging 20 realizations.

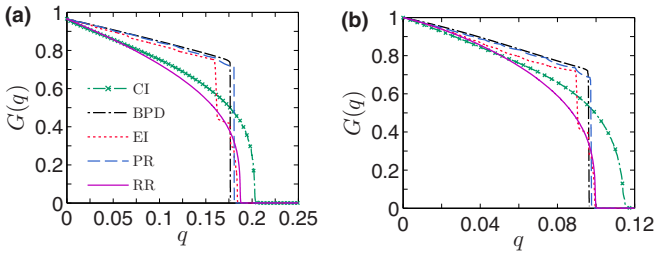


FIG. 4. The fraction $G(q)$ of the size of the largest cluster versus the fraction of removed nodes $q$ (over 50 sample networks) for CI, BPD, EI, PR, and RR on (a) ER networks with $\langle k\rangle = 3.5$ and $n = 10^6$, and (b) SF networks with $\gamma = 3.0$, $\langle k\rangle = 4.0$, and $n = 10^6$.

reason why this happens is ascribed to the fact that $k_{v_i}^{(\mathrm{eff})}$ (used to measure the spreading ability of a node [19]) is harder and harder to identify the nodes with similar degree as $\langle k\rangle$ rising,

$$k_{v_i}^{(\mathrm{eff})} = k_{v_i} - L_{v_i} - M_{v_i}\left(\left\{k_{v_j}^{(\mathrm{eff})} \middle| v_j \in \Gamma(v_i)\right\}\right), \qquad (15)$$

where $\Gamma(v_i)$ consists of all $v_i$'s nearest neighbors, $L_{v_i}$ is the number of leaves (nodes with degree 1) in $\Gamma(v_i)$ and $M_{v_i}(\{k_{v_j}^{(\mathrm{eff})}|v_j \in \Gamma(v_i)\})$ is the number of strong hubs (nodes with $k_{v_j}^{(\mathrm{eff})} \geqslant K$). In other words, more and more nodes have a degree larger than $K$ when the network becomes dense. Hence, we also report the results of EI with $K = \langle k\rangle + 2$ (EI$^2$) in Fig. 5 (but this adaptation is invalid for real-world networks). To summarize: considering the threshold, PR performs better than both CI and EI but slightly worse [$(q_c^{\mathrm{PR}} - q_c^{\mathrm{BPD}})/q_c^{\mathrm{PR}} < 2.58\%$ for all cases] than BPD in the model networks. In contrast to this, RR obtains a quite small $F$ compared to other methods, e.g., $(F^{\mathrm{CI}} - F^{\mathrm{RR}})/F^{\mathrm{CI}} \approx 8.25\%$ in Fig. 4(a) and 10.10% in Fig. 4(b), respectively.

The different performances of BPD in model and real-world networks arouse our interest in another question: how do the loops influence the effectiveness of BPD, since the belief propagation (BP) algorithm is actually sensitive to the existence of circles in a network and most of the real-world networks have a lot of loops (see Table I)? We still employ the paradigmatic ER and SF models to construct our basis networks. Then, for each network, the following strategies are used to enhance the clustering coefficients, i.e., increase the local loops. (i) Randomly choose one node $v_i$ and its two corresponding neighbors $v_j$ and $v_k$ subject to $(v_j, v_k) = 0$, which means that there is no edge between $v_j$ and $v_k$, namely, $(v_i, v_j) = (v_i, v_k) = 1$ and $v_j \neq v_k$. (ii) In the same

TABLE II. The threshold $q_c$ ($G(S;q) < 0.01$), the average giant fraction $F$ and the size of the feedback vertex set $n_{\mathrm{FVS}}$ of HD, CI[a], BPD[a], EI, PR, and RR on the 17 real-world networks. Here CI is with $\ell = 3$ for the Email-EuAll network and $\ell = 2$ for the as-Skitter network. Each result of EI, PR, and RR is obtained by averaging 20 independent realizations. The bold numbers are the minimal value of each objective among these methods for a same network.

| | $q_c \times n$ | | | | | | $F$ | | | | $n_{\mathrm{FVS}}$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Networks[a] | HD | CI | BPD | EI | PR | RR | CI | EI | PR | RR | BPD | PR | RR |
| Power | 975 | 570 | 316 | 337.10 | 440.90 | **282.55** | 0.0449 | 0.0112 | 0.0154 | **0.0076** | 516 | **485.60** | 487.65 |
| CA-GrQc | 912 | 1760 | 398 | 428.25 | 390.20 | **372.10** | 0.0527 | 0.0347 | 0.0356 | **0.0289** | 1449 | **1426.20** | 1427.00 |
| p2p-Gnutella08 | 2045 | 1444 | **1300** | 1508.95 | 1331.20 | 1372.55 | 0.1415 | 0.1651 | 0.1486 | **0.1386** | **1256** | 1276.85 | 1281.00 |
| as-733 | 243 | 192 | 162 | 169.35 | 187.80 | **152.85** | 0.0150 | 0.0097 | 0.0117 | **0.0087** | 216 | **208.00** | 208.60 |
| Scottish | 877 | 2036 | 434 | 471.05 | **432.85** | 442.70 | 0.0542 | 0.0259 | 0.0256 | **0.0231** | 444 | **436.35** | 438.00 |
| CA-AstroPh | 8544 | 4865 | 4198 | 4320.60 | 4055.60 | **4013.10** | 0.1562 | 0.1579 | 0.1368 | **0.1200** | 8626 | 8529.65 | **8525.80** |
| CA-CondMat | 5726 | 3217 | 2569 | 2700.80 | 2559.30 | **2534.35** | 0.0832 | 0.0774 | 0.0694 | **0.0625** | 8323 | 8230.20 | **8228.40** |
| hep-th | 18 097 | 11 184 | 10 294 | 11 002.85 | 9913.35 | **9732.10** | 0.2541 | 0.2742 | 0.2437 | **0.1915** | 12 344 | **12 097.45** | 12 103.15 |
| Cit-HepPh | 22 533 | 14 164 | 13 455 | 14 498.90 | 13 089.05 | **12 982.90** | 0.2645 | 0.2860 | 0.2533 | **0.2056** | 15 405 | **15 133.45** | 15 139.80 |
| Email-Enron | 4097 | 3074 | 2621 | 2764.35 | 2619.00 | **2572.90** | 0.0292 | 0.0314 | 0.0263 | **0.0217** | 7853 | 7748.70 | **7746.35** |
| loc-Gowalla | 53 828 | 31 386 | 26 951 | 26 916.70 | 25 703.10 | **25 015.30** | 0.0868 | 0.0916 | 0.0812 | **0.0625** | 38 841 | **37 690.20** | 37 739.00 |
| Email-EuAll | 1431 | 1193 | **1064** | 6985.80 | 1104.30 | 1077.20 | 0.0056 | 0.0019 | 0.0012 | **0.0008** | 1187 | **1182.80** | 1193.65 |
| com-Amazon | 78 308 | 42 108 | 29 572 | 27 471.15 | 28 056.55 | **26 342.10** | 0.0793 | 0.0619 | 0.0583 | **0.0424** | 85 274 | 82 364.55 | **82 263.80** |
| web-Google | 253 099 | 82 525 | 50 861 | 41 948.85 | 41 175.95 | **33 573.35** | 0.0526 | 0.0322 | 0.0312 | **0.0227** | 208 876 | **205 231.85** | 20 5435.45 |
| PAroad | 273 899 | 71 134 | 21 172 | 17 204.05 | 11 150.15 | **10 124.80** | 0.0417 | 0.0034 | 0.0019 | **0.0012** | 194 443 | **176 535.00** | 177 536.80 |
| Txroad | 307 413 | 82 744 | 20 873 | 16 800.10 | 10 676.50 | **9365.95** | 0.0342 | 0.0019 | 0.0011 | **0.0007** | 239 909 | **217 066.25** | 217 823.05 |
| as-Skitter | 322 128 | 151 846 | 74 286 | 70 901.00 | **62 059.25** | 63 977.35 | 0.0394 | 0.0287 | 0.0239 | **0.0215** | 228 775 | **224 356.65** | 22 5329.90 |

[a]The source code of CI is from Ref. [43]. The source code of BPD is from Ref. [44].

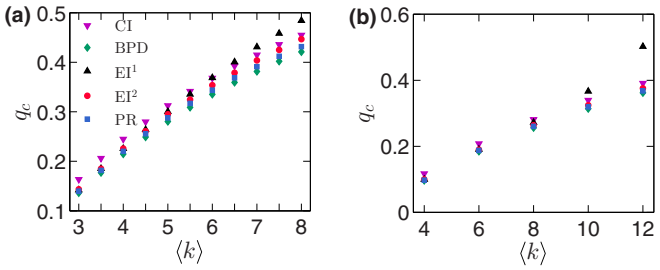


FIG. 5. The threshold $q_c$ in dependence on the average degree $\langle k \rangle$ (50 sample networks for each $\langle k \rangle$) for CI, BPD, EI[1], EI[2], and PR for (a) ER networks with $n = 10^5$ and (b) SF networks with $\gamma = 3.0$ and $n = 10^5$.

way to respectively choose one of the neighbors of $v_j$ and $v_k$, assuming they are $v_{jj}$ and $v_{kk}$ satisfying $(v_j, v_{jj}) = (v_k, v_{kk}) = 1$, $(v_{jj}, v_{kk}) = 0$, and $v_{jj} \neq v_{kk}$. (iii) Cut (delete) the edges $(v_j, v_{jj})$ and $(v_k, v_{kk})$ and at the same time add two new edges $(v_j, v_k) = (v_{jj}, v_{kk}) = 1$. (iv) repeat (i)-(iii) until the network reaches our demand, i.e., a given clustering coefficient. In this manner, the clustering coefficients of these networks can be improved and, apparently, the degree distribution of them is kept constant. As illustrated in Fig. 6, the fraction $n_{\mathrm{FVS}}/n$ of PR rise more slowly than BPD with the increase of the clustering coefficients CC in both ER and SF networks, while the threshold $q_c$ of PR decreases more quickly than BPD. This may indicate that PR is more suitable than BPD for real-world networks. Therefore, we also show the performances of BPD, PR, and RR for the FVS problem in Table II where PR finds a smaller FVS than BPD in almost all the networks (16/17).

Moreover, we consider the two largest networks, the TXroad network (with maximal degree 12) and the as-Skitter network (with maximal degree 35 455), to demonstrate the efficiency of the proposed methods. Since it is hard to analyze the computational complexity of RR and PR in detail, we here put them as well as CI and BPD (open-source codes written by either C or C++ program) in the same simulated environment and compare their time consumptions. As illustrated in Fig. 7, both PR and RR get smaller thresholds than CI and BPD within a quite short time, in particular, RR takes only 3.6 s to obtain a better result than CI and BPD in the TXroad network. Note that the running time of CI and BPD reported here may be as a reference but not as a standard.

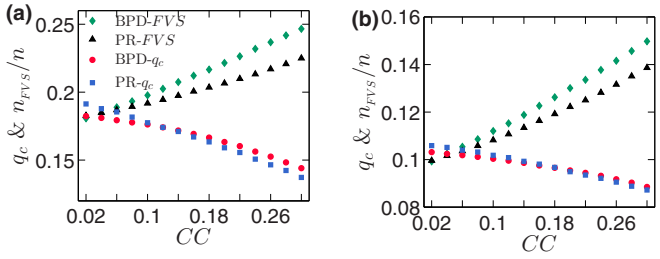


FIG. 6. The threshold $q_c$ and the fraction $n_{\mathrm{FVS}}/n$ of the feedback vertex set versus the clustering coefficients CC for BPD and PR in (a) ER networks with $n = 10^4$ and $\langle k \rangle = 3.5$, and (b) SF networks with $\gamma = 3.0$, $\langle k \rangle = 4.0$, and $n = 10^4$.

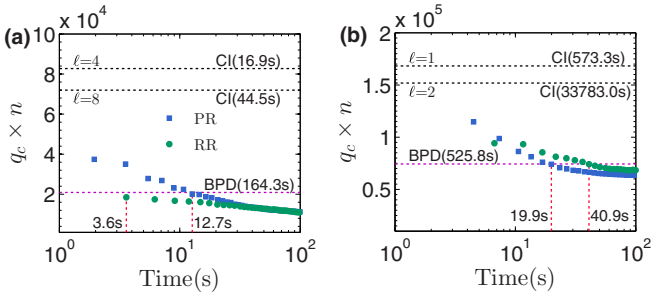


FIG. 7. The running time [measured by second (s)] of CI, BPD, PR (with $\tau_s = 5$) and RR (with $\tau_s = 10$) on (a) the TXroad network and (b) the as-Skitter network. The horizontal dash lines correspond to the thresholds of either CI or BPD. The values marked beside the vertical dash lines are related to the computational time indicating that the proposed methods begin to have smaller thresholds than both CI and BPD. All the results are obtained by averaging 20 implementations.

Finally, the susceptible-infectious-recovery (SIR) epidemic spreading model [5,18,46] is used to investigate the spreading process of a virus on the email-Enron network and the loc-Gowalla network by comparing the CI, EI, and RR methods. For a given network under SIR simulation, its nodes belong to either the susceptible, infected, or recovered state. And before the start of the simulation, a part of nodes are previously identified and removed from the network based on a certain strategy. Then one random node is selected from the remaining network as the infected source and the others are to be susceptible. In each time step, the infected nodes infect their susceptible neighbors with the infection rate $\lambda$, and then they recover with rate $\eta$. The recovered nodes are removed from the network too. This process is repeated until there is no infected node in the network. The simulation results are shown in Fig. 8 where $\lambda$ and $\eta$ are fixed to 0.2 and 0.05, respectively. On both networks, RR has a significantly lower value (9.5 to 20.0 times) of recovered individuals than EI under the same immunized fraction $q$ [Figs. 8(a) and 8(b)]. Considering the final recovered fraction $R_f$ [Figs. 8(c) and 8(d)], RR also outperforms CI and EI in almost all situations.

## IV. CONCLUSION

In this paper, two methods as effective strategies have been developed for the robustness and target immunization problems based on percolation transition. The proposed strategies choose the removed (immunized) fraction by repeatedly investigating and capturing the interrelationship among nodes. To evaluate the effectiveness of both proposed methods, we conduct numerous simulations on two types of model networks as well as 17 real-world networks from different fields. The results, especially most of the empirical networks, clearly illustrate that our strategy considerably outperforms the existing well-known strategies, like CI [3] and EI [19]. In addition, our strategies might open up a new path to investigate more effective solutions to the robustness and immunization problems as well as obtain the minimal feedback set [4] in network science.

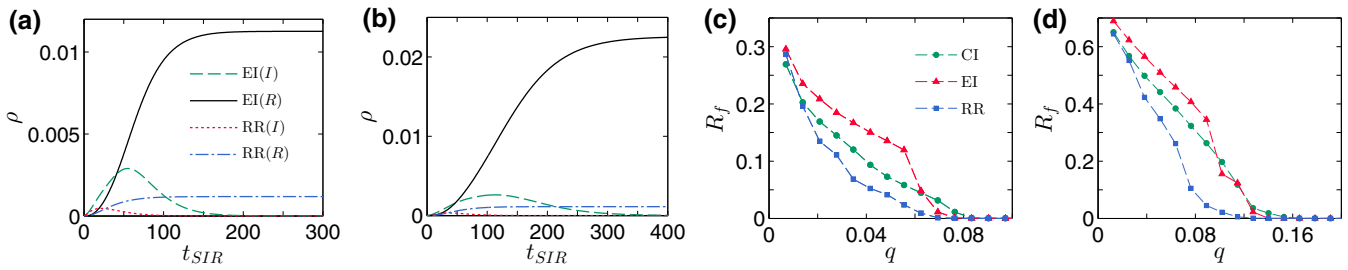


FIG. 8. The SIR simulation results of CI, EI and RR respectively on the Email-Enron network (a, c) and the loc-Gowalla network (b, d), including (a–b) the rate $\rho$ of infected ($I$) and recovered ($R$) individuals versus the spreading time step $t_{\mathrm{SIR}}$ under the immunized fraction $q = q_c^{\mathrm{RR}}$, and (c–d) the final recovered fraction $R_f$ versus the fraction of immunized nodes $q$. In each network, $10^4$ independent selections are conducted.


## ACKNOWLEDGMENTS

The authors would like to acknowledge the European Regional Development Fund (ERDF), the German Federal Ministry of Education and Research, and the Land Brandenburg for supporting this project by providing resources on the high-performance computer system at the Potsdam Institute for Climate Impact Research. Y.L. gratefully acknowledges support from a China Scholarship Council (CSC) scholarship and J.K. for the support of the Russian Science Foundation (Grant No. 16-12-10198).



[1] R. Albert, H. Jeong, and A.-L. Barabási, Nature (London) **406**, 378 (2000).
[2] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. Lett. **85**, 5468 (2000).
[3] F. Morone and H. A. Makse, Nature (London) **524**, 65 (2015).
[4] S. Mugisha and H.-J. Zhou, Phys. Rev. E **94**, 012305 (2016).
[5] Z. Wang, C. T. Bauch, S. Bhattacharyya, A. d'Onofrio, P. Manfredi, M. Perc, N. Perra, M. Salathé, and D. Zhao, Phys. Rep. **664**, 1 (2016).
[6] R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin, Phys. Rev. Lett. **86**, 3682 (2001).
[7] R. Cohen, S. Havlin, and D. ben-Avraham, Phys. Rev. Lett. **91**, 247901 (2003).
[8] P. Wang, M. C. González, C. A. Hidalgo, and A.-L. Barabási, Science **324**, 1071 (2009).
[9] D. Brockmann and D. Helbing, Science **342**, 1337 (2013).
[10] F. Peruani and G. J. Sibona, Phys. Rev. Lett. **100**, 168103 (2008).
[11] F. Peruani and L. Tabourier, PloS ONE **6**, e28860 (2011).
[12] A. D. Kramer, J. E. Guillory, and J. T. Hancock, Proc. Natl. Acad. Sci. USA **111**, 8788 (2014).
[13] C. M. Schneider, A. A. Moreira, J. S. Andrade, S. Havlin, and H. J. Herrmann, Proc. Natl. Acad. Sci. USA **108**, 3838 (2011).
[14] D. Li, B. Fu, Y. Wang, G. Lu, Y. Berezin, H. E. Stanley, and S. Havlin, Proc. Natl. Acad. Sci. USA **112**, 669 (2015).
[15] R. Pastor-Satorras and A. Vespignani, Phys. Rev. E **65**, 036104 (2002).
[16] L. K. Gallos, F. Liljeros, P. Argyrakis, A. Bunde, and S. Havlin, Phys. Rev. E **75**, 045104 (2007).
[17] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, Phys. Rev. E **65**, 056109 (2002).
[18] Y. Chen, G. Paul, S. Havlin, F. Liljeros, and H. E. Stanley, Phys. Rev. Lett. **101**, 058701 (2008).
[19] P. Clusella, P. Grassberger, F. J. Pérez-Reche, and A. Politi, Phys. Rev. Lett. **117**, 208301 (2016).
[20] K. Gong, M. Tang, P. M. Hui, H. F. Zhang, D. Younghae, and Y.-C. Lai, PLoS ONE **8**, e83489 (2013).
[21] W. Wang, M. Tang, H.-F. Zhang, H. Gao, Y. Do, and Z.-H. Liu, Phys. Rev. E **90**, 042803 (2014).
[22] Y. Liu, Y. Deng, and B. Wei, Chaos **26**, 013106 (2016).
[23] T. Bian and Y. Deng, Chaos **28**, 043109 (2018).
[24] M. E. J. Newman, Phys. Rev. Lett. **89**, 208701 (2002).
[25] L. C. Freeman, Sociometry **40**, 35 (1977).
[26] H.-J. Zhou, Eur. Phys. J. B **86**, 455 (2013).
[27] C. M. Schneider, T. Mihaljev, and H. J. Herrmann, EPL (Europhys. Lett.) **98**, 46002 (2012).
[28] Y. Liu, B. Wei, Z. Wang, and Y. Deng, Phys. Lett. A **379**, 2795 (2015).
[29] D. Achlioptas, R. M. D'Souza, and J. Spencer, Science **323**, 1453 (2009).
[30] S.-M. Qin and H.-J. Zhou, Eur. Phys. J. B **87**, 273 (2014).
[31] L. Zdeborová, P. Zhang, and H.-J. Zhou, Sci. Rep. **6**, 37954 (2016).
[32] D. J. Watts and S. H. Strogatz, Nature (London) **393**, 440 (1998).
[33] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[34] J. Leskovec, J. Kleinberg, and C. Faloutsos, ACM Trans. Knowl. Disc. Data (TKDD) **1**, 2 (2007).
[35] R. Matei, A. Iamnitchi, and P. Foster, IEEE Internet Comput. **6**, 50 (2002).
[36] J. Leskovec, J. Kleinberg, and C. Faloutsos, in *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (ACM, Chicago, Illinois, USA, 2005), pp. 177–187.
[37] J. Gehrke, P. Ginsparg, and J. Kleinberg, ACM SIGKDD Explorations Newsletter **5**, 149 (2003).
[38] B. Klimt and Y. Yang, CEAS 2004 - First Conference on Email and Anti-Spam, July 30-31, 2004, Mountain View, California, USA (2004), https://dblp.uni-trier.de/db/conf/ceas/ceas2004.html.
[39] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, Internet Math. **6**, 29 (2009).
[40] E. Cho, S. A. Myers, and J. Leskovec, in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge*

*Discovery and Data Mining* (ACM, San Diego, California, USA, 2011), pp. 1082–1090.
[41] J. Yang and J. Leskovec, Knowl. Inf. Syst. **42**, 181 (2015).
[42] A. Srivastava, B. Mitra, N. Ganguly, and F. Peruani, Phys. Rev. E **86**, 036106 (2012).
[43] http://www-levich.engr.ccny.cuny.edu/webpage/hmakse/software-and-data/.
[44] http://power.itp.ac.cn/∼zhouhj/codes.html.
[45] P. Erdős and A. Rényi, Publ. Math. (Debrecen) **6**, 290 (1959).
[46] M. E. J. Newman, Phys. Rev. E **66**, 016128 (2002).