IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 1, 2018, accepted May 28, 2018, date of publication June 4, 2018, date of current version June 20, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2843532

Identification of Vital Nodes in Complex Network via Belief Propagation and Node Reinsertion

JILONG ZHONG[1,2,3], FENGMING ZHANG[1], AND ZHENGXIN LI[1]

[1]Equipment Management and UVA Engineering College, Air Force Engineering University, Xi'an 710051, China
[2]School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China
[3]Science and Technology on Reliability and Environmental Engineering Laboratory, Beihang University, Beijing 100191, China

Corresponding author: Fengming Zhang (z_fengming@163.com)

This work was supported by the National Natural Science Foundation of China Grant 71771009, Grant 71501185, and Grant 61502521.

ABSTRACT Vital nodes play a pivotal part of network structure and dynamics, where finding the minimal size of a set of vital nodes belongs to an NP-hard problem and cannot be solved by a polynomial algorithm. Recent studies of vital nodes identification mostly rely on the structural information, such as collective influence and degree. However, the performance of local-based methods varies for different structure, while the complexities of global-based methods are generally high for most situations. Here, we map the problem into an optimization issue based on global information of network structure and propose a belief propagation and node reinsertion (BPR) method with almost linear time complexity, where finding the minimum feeding back vertex set is a key. Compared with several state-of-the-art heuristic methods, the BPR method has advantages of high accuracy and practicability of vital nodes identification and low computational complexity. Under two attack schemes: static and dynamical, extensive experiments of Erdős–Rényi and scale-free models and real-world networks of the power grid and traffic network convincingly demonstrate that the BPR method remarkably outperforms other methods in vital nodes identification. This helps to reassess the operational risk of a network and improve robustness ranging from network design schemes, protection strategies to failure mitigation.

INDEX TERMS Complex network, attack vulnerability, node reinsertion, belief propagation.

I. INTRODUCTION

Attack vulnerability in the complex network has recently attracted considerable attention, where many efforts have been devoted to focus on dynamics such as cascading failure, virus spreading, and information exchanging [1]–[5]. Originated from the physically connected computer network, attack vulnerability can be characterized as the decrease of network performance when nodes or edges are removed. As vulnerabilities of many infrastructures stem from the existence of vital nodes which can be vulnerable parts of a network under malicious attack, gaining an insight into the effect of removal of such vital nodes is critical. Although there are not many vital nodes in a network, their impacts can spread through the whole network. Evidence has demonstrated that even a locally intentional attack on a small number of nodes (edges) can lead to a serious global system collapse. For example, 2003 major blackout in north America caused a large swath of districts to be paralyzed and resulted in more than $4 billion financial losses, originated from outages of several critical transition lines [6]. Hence, it is critical to identify vital nodes in a network, which can help to improve the ability of a network to defend against attacks. The faster the network will crash into numerous components with fewer numbers of nodes, the better the algorithm.

In the past decades, questions regarding vital nodes identification are primarily asked about (i) What are vital nodes? (ii) How to identify these nodes? (iii) What is the impact of the removal of vital nodes? Numerous notable examples of network studies, ranging from reliability engineering, disaster risk science, social science, physics, and biology, make efforts to identify vital nodes that most crucially affect network topology and functionality [7]–[13]. Researchers have devoted efforts towards investigating how network robustness changes when vital nodes are removed from the network according to distinct centrality measures [14], [15]. Based on heuristic methods, recent studies of vital nodes identification can be classified into three categories for their different methodologies. Some researchers use neighbor-based local information of a network such as the Collective Influence (CI) [3], the Highest Degree (HD) [16], K-Core [17] and new K-core [18], and LocalRank methods [19]. While the time and space complexity of these local-based methods are low, the performances of them vary in different situations. The second type of study pays attention to path-based global information of a network, where Gradient maximum likelihood algorithm [20], Betweenness centrality [21], Closeness centrality [22] and Kats centrality [23] are some recently typical methods that mainly consider paths of network flow. This type of method usually has a better result, whereas the complexities of this type are usually high and not suitable for calculation of a network with large scale. The third type of method quantifies the importance of nodes through eigenvector of a network considering mutual enhance effect between nodes. This type of method is typically iterative algorithms that are different from the former two class of methods, including eigenvector centrality [16], PageRank [24], LeadRank algorithm [25], and Hub centrality [26].

These methods are widely used in vital nodes identification. However, they characterize limited parts of what it implies for a node to be critical in a network and each has its own shortcoming and limitation [27]. This may lead to underestimating the importance of some nodes. Moreover, it is a known fact that targeting vital nodes in a network is a nondeterministic polynomial hard (NP-hard) problem and cannot be solved by a polynomial algorithm. Therefore, it is rather difficult to calculate the accurate analytic solution. Here, we regard it as a combinatorial optimization problem and deal with the issue by a method based on global information of network structure: Firstly, we aim to find the minimum feeding back vertex set, which can be regarded as a problem of optimal attack based on message-passing theory [28], [29]; Secondly, optimize the attack order of vital nodes obtained from step one by using two different node reinsertion methods proposed in this paper. Compared with other methods, the simulation results show that the BPR algorithm achieves better performance in both speed and accuracy.

The rest of the paper is organized as follows. We first give a brief review of details of some recently proposed benchmark centrality measures. Then, the BPR algorithm is described in detail in Section III. In Section IV, vulnerability metric and network information are also given, while numerical results are mainly given in Section V. In order to compare BPR and other typical measures, we first put forward a method correlation analysis. In what follows, the performance of BPR measure and other benchmark measures are analyzed under both static and dynamical strategies, where we collected data of six networks including model and real networks. Moreover, we further investigate the computational efficiency of different methods. At the end of this paper, some conclusions are presented in Section IV.

II. BRIEF REVIEW ON TRADITIONAL METHODS

Here, we assume the structure of a network is unweighted and undirected without multiple edges and self-loops. A network can be represented as $G = (V, E)$, where $N = |V|$ is the number of nodes and $M = |E|$ is the number of edges in a network. If a vertex set $v \in V$ is removed from a network, the whole graph may be separated into a number of independent components, referring to an attack event for the network.

A. COLLECTIVE INFLUENCE

Recently, a novel efficient Collective Influence (CI) centrality is published in Nature by mapping influence maximization problem into an optimal percolation problem [3]. Based on the local structural information, CI is calculated by

$$CI(i) = (k_i - 1) \sum_{j \in \partial Ball(i,l)} \left(k_j - 1\right) \tag{1}$$

where k_i and k_j are the degree of node i and node j, and $j \in \partial Ball(i, l)$ represents that node j belongs to the neighborhood of node i from a distance l.

B. BETWEENNESS

In general, there is more than one shortest path between arbitrarily chosen two nodes in a network. Betweenness centrality is a measure that counts how many shortest paths that go through a node [21]. Nodes with a high betweenness centrality will be more influential than other nodes with low betweenness.

$$Bet(i) = \sum_{i \neq s, i \neq t, s \neq t} l_{st}^{i} / l_{st} \tag{2}$$

where i, s, and t are arbitrarily chosen nodes in a network. l_{st} represents the number of paths between node s and t, whereas l_{st}^{i} is the number of paths going through node i.

C. THE HIGHEST DEGREE

The Highest Degree centrality is one of the most straightforward methods based on the degree of a node. The highest degree centrality can be defined as

$$HD(i) = k_i / (n - 1) \tag{3}$$

where k_i is degree of node i and $HD(i)$ is a normalized degree centrality.

D. K-CORE

K-core centrality emphasizes the importance of the location of a node, which is a better indicator than the degree centrality for evaluating the spread influence of a node [17]. For example, at the first step, all the nodes with degree $k \leq 1$ should be removed from the network until the degrees of all the remaining nodes satisfy $k > 1$ and the removal nodes belong to shell one. Then, we iteratively remove all the nodes with degree $k \leq 2$, the rest can be done in the same manner.

E. PAGERANK

PageRank centrality supposes the importance of a node is determined by both quantity and quality of the nodes pointed to it, which does not only consider an important node pointing to another node but takes a diluted effect of a prestigious node on others into account [24]. It can be defined as

$$PR(i) = \alpha \sum_{j} A_{ij} x_j / k_j^{out} + \beta \tag{4}$$

where α and β are positive constants, A_{ij} is an adjacent matrix, x_j is the neighborhood of node i, and k_j^{out} is the outdegree of node j.

III. BELIEF PROPAGATION AND NODE REINSERTION (BPR) METHOD

Here, BPR algorithm is proposed to obtain a faster way to break down a network, which mainly consists of two steps: belief propagation and node reinsertion. In this BPR algorithm, key nodes are calculated by belief propagation, while the attack sequence is optimized by node reinsertion process. Moreover, two types of different node reinsertion methods are also put forward.

A. BELIEF PROPAGATION (BP)

As most of the local-based algorithms do not take the global structure of a network into consideration, the performance of methods will be limited. Based on distributed message-passing theory [28], the belief propagation method is used to identify vital nodes with almost a linear time complexity $O(N \ln N)$. Compared with other global methods that identify vital nodes by the number of the shortest paths that go through a node, BPR method transforms the problem into a combination optimization issue of finding the minimum feedback vertex set (FVS) for a better accuracy. FVS is a node set that intersects with every loop of the network and if removed, all loops break and leave behind a forest in the network. It is known that in sparse networks the small connected components are mostly trees. In this way, the removal of the root node of the remaining forest may induce an abrupt collapse of the whole network very efficiently. Hence, vital nodes can be regarded as root nodes and the minimum feedback vertex set [30]. FVS problem is also an NP-hard problem and can be approximately solved based on mean-field theory in statistical physics. Here we introduce how to apply belief propagation method to FVS problem.

The microscopic configuration of a network can be denoted as $\{A_1, A_2, \ldots\ldots, A_N\}$, where A_i represents the state variable of a node i and can only take a value of $A_i = 0, A_i = i \; or \; A_i = j (j \in \partial i)$. If node i is unoccupied or removed, the state $A_i = 0$. If $A_i = i$, it indicates that node i is occupied and itself is a parent node for its neighborhood, whereas $A_i = j$, it means the parent of node i is node j and node i is also occupied.

According to replica-symmetric mean field theory [28], [30], the belief propagation can be represented as self-consistent equations

$$q_{i\to j}^{0} = \frac{1}{z_{i\to j}} \tag{5}$$

$$q_{i\to j}^{i} = e^{xw_i} \prod_{k\in\partial i\backslash j} (q_{k\to i}^{0} + q_{k\to i}^{k})/z_{i\to j} \tag{6}$$

$$q_{i\to j}^{l} = e^{xw_i}\left(1 - q_{l\to i}^{0}\right) \prod_{m\in\partial i\backslash j,l} (q_{m\to i}^{0} + q_{m\to i}^{m})/z_{i\to j}, \quad l \in \partial i\backslash j \tag{7}$$

where $q_{i\to j}^{A_i}$ is the marginal state probability of node i when its neighbor node $j \in \partial i$ is removed, and both x, w are the weighting parameters in the equation. Hence, the BP Eqs (5), (6), and (7) provide all the three state probabilities of a node. Note that $z_{i\to j}$ is a normalization factor denoted as

$$z_{i\to j} \equiv 1 + e^{xw_i}[\prod_{k\in\partial i\backslash j} (q_{k\to i}^{0} + q_{k\to i}^{k}) + \sum_{k\in\partial i\backslash j} \left(1 - q_{l\to i}^{0}\right) \prod_{m\in\partial i\backslash j,l} (q_{m\to i}^{0} + q_{m\to i}^{m})] \tag{8}$$

Eqs (5)-(8) form the common self-consistent belief propagation method. We can iteratively calculate these equations for a network until the results converge to a fixed value. The node with the highest $q_{i\to j}^{0}$ probability will be removed from the graph along with its entire attached links. The algorithm stops after no loops exist in the graph, where there are only trees or forest in the network topology. Then, an appropriate node, i.e. the root node in the graph, is supposed to be deleted in the largest tree until the graph completely breaks down.

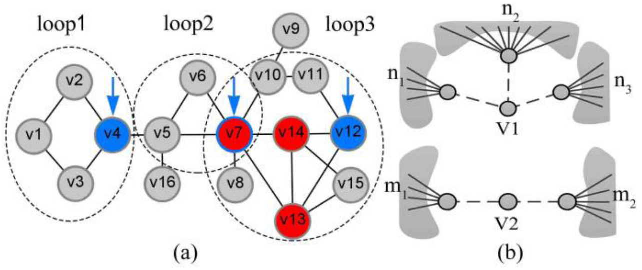


FIGURE 1. Schematic diagram of BPR model. (a) Illustration of BP method on a network (b) Node reinsertion diagram.

An example of BP method is shown in Fig. 1a. It is found that node 7, node 14, and node 13 have a relatively higher degree and will be firstly removed by the highest degree centrality. After removing these three nodes, the largest connected component still contains 7 nodes. As the main goal of the BP method is to remove loops in a network, which will generally result in a smaller size of the component. In Fig1a, node 4 in loop$_1$, node 7 in loop$_2$ and node 12 in loop$_3$ are considered to be firstly removed by the BP method. Accordingly, the number of nodes in the largest component will decrease to 3 by contrast.

To describe the belief propagation algorithm in detail, we give a pseudo-code in Table 1. We first input a graph with adjacency list, and then initialize two weight parameters. The next is the core part of the algorithm, which is called BP iteration. According to Eqs (5)-(8), we can calculate the empty probability q_0 of each node based on current inputting messages. By sorting the empty probability q_0 from large to small, we add these nodes to the removing list until the size of the biggest component in graph decreases lower than the threshold. Here, the threshold is a certain small value

TABLE 1. Pseudo-code of belief propagation.

```
Algorithm1: Belief Propagation

INPUT: adjacency list graph G
OUTPUT: attack list lr;
set graph G=(V,E);
set weight1 x=11, weight2 w=1;
set comp_thresh=0.01;
set bigcomp_size=the largest component size of G;
while bigcomp_size > comp_thresh do
    for each active node i do
        temp_a = 1;
        temp_b = 0;
        temp_c = exp(-x*w);
        for each neighbor j of node i do
            if state[j]=="ON" then
                q_0 = j.q_0;   % probability of node j to be removed
                    q_r = j.q_r;   % probability of node j to be a root
                    q0_r = q_0 + q_root;
                temp_b = temp_a * (1 - q_0) + temp_b * q0_r;
                temp_a *= q0_r;
                max = max(temp_a, temp_b);
                if max<temp_c then
                    max = temp_c;
                end if
                temp_a /= max;
                temp_b /= max;
                temp_c /= max;
            end if
        end
        normlization = temp_a + temp_b + temp_c;
        q_0 = temp_c / normlization;
        i.q_0=q_0;
        q0_list[i]=q_0;
    end
    sort q0_list from large to small;
    add q0_list[0] to lr % add node with max probability to be removed
    set state[q0_list[0]] = "OFF"
    update  bigcomp_size;
end
```

(e.g. thresh=0.01 or even small), which is used to determine the relative size of the giant component whether decreases to a given size to stop the algorithm. At this point, BP process terminates when no loops are present and output the resulting attack set of the original graph.

B. NODE REINSERTION (R)

Usually, attack order and the set of vital nodes are both determined by the centrality of nodes in a network. Nevertheless, it ignores the situation in which a minor perturbation to several nodes with relatively low centrality at the same time can also result in large-scale damages. In other words, it is important to recalculate and optimize the attack order even though we have already obtained the vital node set. Node reinsertion adds back a finite fraction of nodes at each step to keep the maximally fragmented network, which is a post-process and refinement of vital nodes identification at an early stage. As described in Fig.1b, two distinct types of reinsertion methods are proposed, where both have their unique advantages for different network topologies. In this algorithm, the complexities of searching, inserting and deleting are all $O(N \ln N)$ for both R1 and R2.

1) THE SMALLEST NUMBER NODE REINSERTION (R1)

For the smallest number node reinsertion, the reinserted nodes are required to join the components with the smallest increasing number of components, called R1.

In other words, we recursively reinsert a node that is in the candidate node set obtained by BP iteration such that each reinserted node connects the least components in the network. Until all the removed nodes are reinserted into the network, the inverse sequence of reinsertion is the final optimal attack order. Fig.1b shows an example on how to reinsert a node into the network. If there are two candidates nodes V_1 and V_2 to be reinserted into the network, node V_2 that connects only two components will be first chosen to keep maximally fragmented network.

2) THE SMALLEST SIZE NODE REINSERTION (R2)

The second type of reinsertion requires the reinserted node to join the components of the smallest sizes, called R2. Other processes of R2 are the same as those of R1. For example, in Fig 1b, if the size of components satisfies $n_1 + n_2 + n_3 < m_1 + m_2$, candidate node V_1 is firstly selected to be reinserted. This difference will have an advantage over networks with distinct topologies.

TABLE 2. Pseudo-code of two types of node reinsertion.

```
Algorithm2: Node Reinsertion (R1 and R2)

INPUT: attack list lr, adjacency list graph G, number of removed nodes
nr;
OUTPUT: the attack order;
set state of node i "OFF" for each node lr, while others are set "ON" ;
set  fitlist  for fitness of each node i in lr to 0;
assign component label for each node i; %Run the algorithm
for each node i in lr do
      set fitlist of node i to 1/(number of components node i would join in
        (when use  NR1)) or  1/(size of component would node i join in
        (when use NR2));
    end
sort fitlist from large to small ;
for each node i in lr    do
    set state of node i "ON";
    assign component label for each node;
    for each i in lr  do  % update fitlist
        if node i in "OFF" state then
            set fitlist of node i to 1/(number of components node i would
            join in (when use  NR1)) or 1/(size of component would node i
            join in (when use NR2));
        end if
    end
    sort fitlist from large to small;
end
reverse rl  % to get attack order
```

Pseudo-code of R1 and R2 are described in detail in Table 2. Both start with the attack list obtained from BP process. State of each node is assigned based on an attack list before running the reinsertion procedure. To achieve the optimal attack order, we iteratively calculate the fitness of each node in the attack list calculated by Eqs(11), where N_i represents the number of components node i would join in

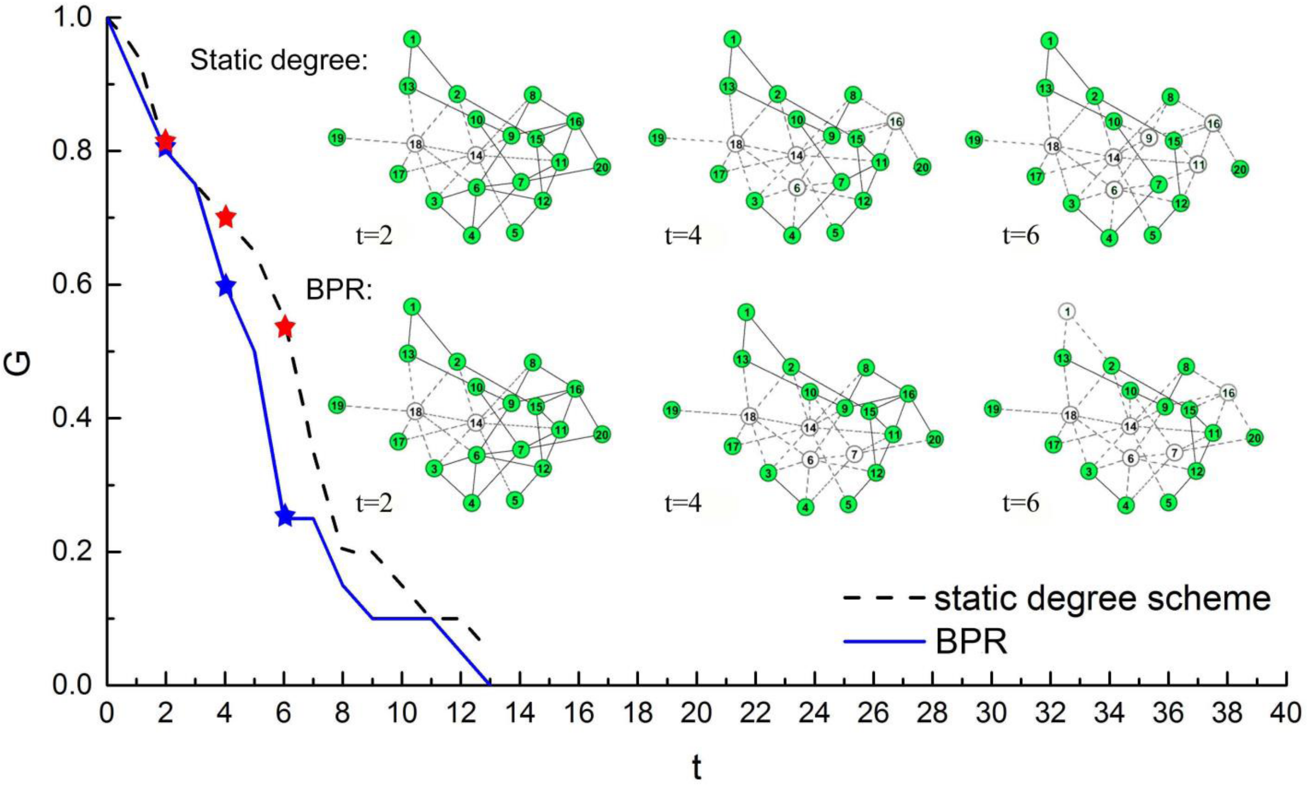


FIGURE 2. Illustration of BPR method compared with static degree method.

and S_i stands for the size of clusters node i would join in. After several iterations, an output of optimal attack list can be obtained.

$$fit_i = \begin{cases} 1/N_i & (R1) \\ 1/S_i & (R2) \end{cases} \tag{9}$$

To further illustrate BPR method, we give an example of comparison between BPR method and static degree method on a small network with 20 nodes and 36 edges shown in Fig. 2. In this case, static degree means degree centrality of each node is calculated only once during the whole process. First, BP procedure with R1 is performed on the graph to get attack list of node set {1, 6, 7, 14, 16, and 18}, while node set {6, 9, 11, 14, 16, and 18} with the highest degree is obtained by static degree method. The attack list calculated by BPR1 is {18 → 14 → 6 → 7 → 16 → 1}, whereas {14 → 18 → 6 → 16 → 9 → 11} is obtained by static degree method. It is found that even a minor modification on the attack order is possible to obtain a considerably different result, where static degree method (in a dashed line) has a worse performance than BPR method (in a solid line).

IV. METRICS AND NETWORK DESCRIPTION

A. METRICS DEFINITION

In general, the robustness of a network is measured by the critical fraction of removed nodes q_c, where network completely falls apart at the value of q_c. However, this measurement cannot indicate the early collapse of a network suffering from a malicious attack. There are numerous other network measurements such as graph entropy [31], network efficiency [32], pairwise connectivity [33] etc. In order to measure the attack vulnerability of a network for each stage, V-index [34] is considered referring to the relative size of the largest component in the network during the whole node removal process, which can be defined as:

$$V = 1 - \left(\sum_{i=1}^{N} \sigma(i/N) \right) /N \tag{10}$$

where V is the attack vulnerability index which can be shown as 1 minus the area surrounded by the curve and axes, N is the total number of nodes, and σ is the proportion of the largest component in a network after removing $q = i/N$ fraction of nodes.

In relation to a complex network, vulnerability refers to the inability to withstand damage caused by the random or malicious attack, which can be quantified as a V-index. Numerous centrality measures have been proposed from different perspectives in previous researches. The centrality of a node can be applied as a basis of attack strategy. In this paper, we will first introduce several state-of-the-art benchmark centrality measures and compare these measures with BPR method to test the attack effects on different networks.

B. NETWORK DESCRIPTION

Two model networks including Erdős–Rényi (ER) network [35] and Scale-Free (SF) network [36] and four real networks of distinct types are introduced to study effects of removal on networks. ER network and SF network are two kinds of models with different topologies, where they have completely different behaviors under random or malicious attack. It is known that the SF network is more vulnerable to malicious attack but appears to be more robust under random attack than ER model. We employ an SF network with power exponent $\gamma = 2.5$ and an ER network with edge existence probability $p = 0.00035$.

The simulation also focuses on some real networks in different fields, including C.elegans neural network [37], the road network of Oldenburg in Germany [38], the power grid of Europe [39], and human protein-protein interaction network [40]. Details of basic network parameters for above networks are described in Table 3. In order to exhibit the topology of different networks, we show the number of nodes $|V|$, edges $|E|$, mean degree $\bar{k}$, clustering coefficient $\bar{c}$, and average path length $\bar{l}$ for each network.

TABLE 3. Basic property of different networks.

Network	$\lvert V\rvert$	$\lvert E\rvert$	$\bar{k}$	$\bar{c}$	$\bar{l}$
SF Network	10,000	20,000	4	0.013	4.41
ER Network	10,000	17,500	3.5	0.00035	7.39
C.elegans	453	4596	20.29	0.646	2.66
Oldenburg	6105	7029	2.30	0.011	40.69
Power Grid	1467	2289	3.12	0.126	17.39
Protein	3133	6726	4.29	0.063	4.84

V. NUMERICAL RESULTS

In this section, attack effects on different networks are investigated by some specific procedures based on the above-mentioned BPR measure and other centrality measures. We calculate and analyze the correlation between these measures. The size of the giant component of each network and its vulnerability with respect to distinct centrality measures are also analyzed by removing nodes in the order of decreasing centrality. In addition, vital nodes to be removed are based on two types of different schemes, i.e. static attack and dynamical attack.

A. CORRELATION ANALYSIS BETWEEN DIFFERENT METHODS

To seek the correlation between BPR method and the other centrality measures, we analyze the Pearson correlation, which is a measure of linear correlation between two vectors described as:

$$r = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \Big/ \left(\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2} \right) \quad (11)$$

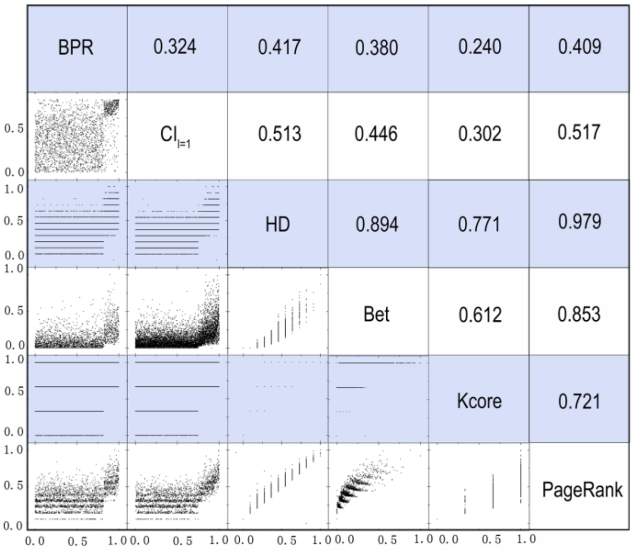


FIGURE 3. Pearson correlation between six distinct centrality measures on an ER network.

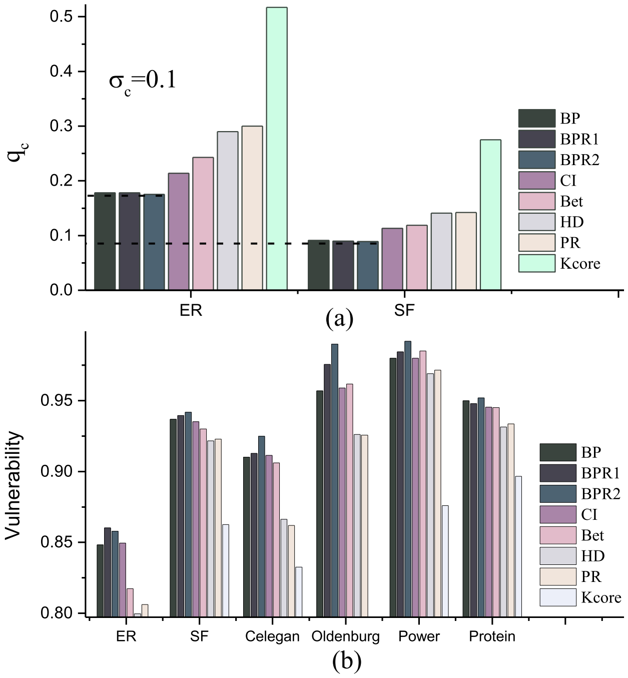
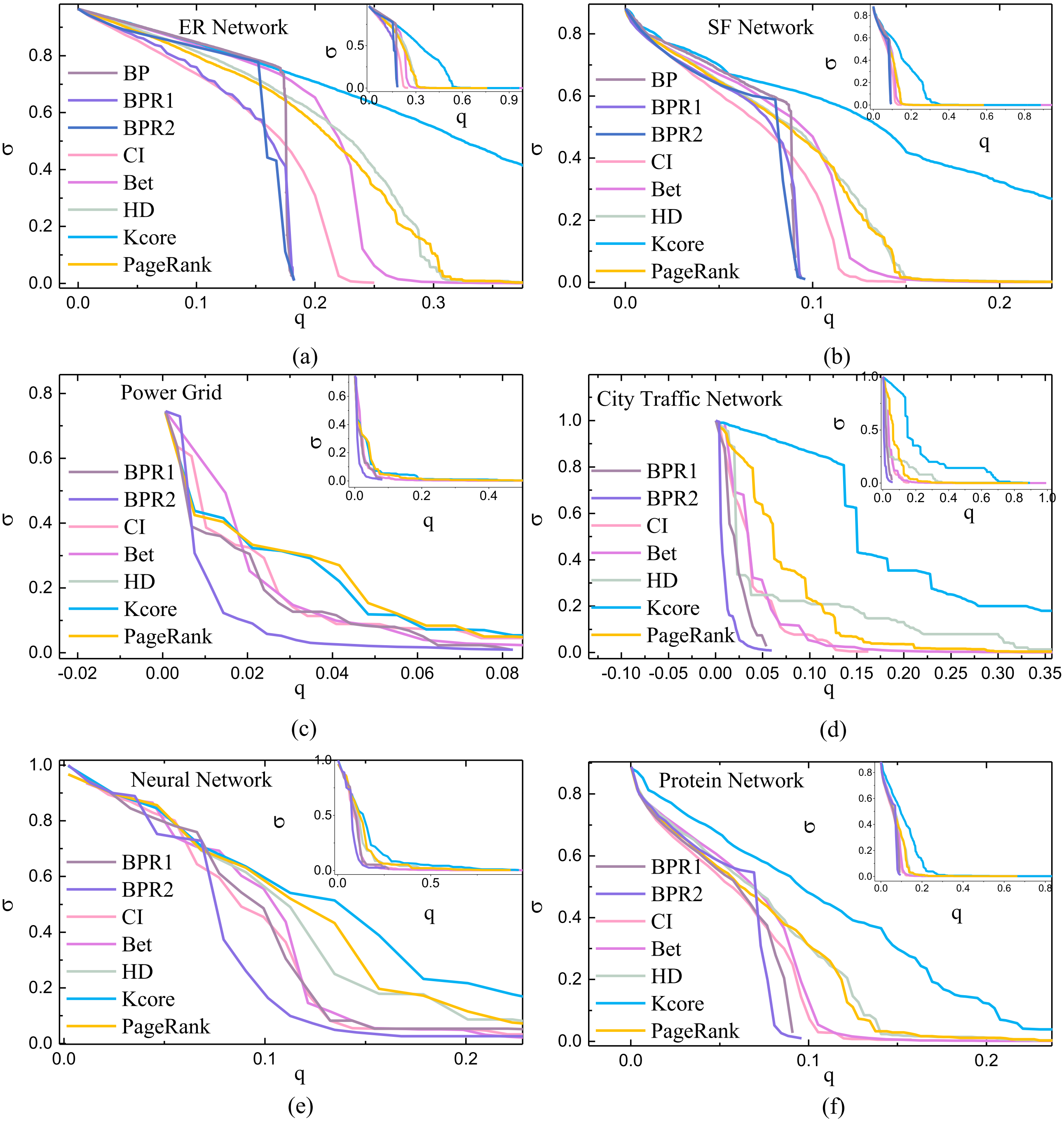


FIGURE 4. Static attack (a) Critical removal fraction q_c for methods of different types. (b) Vulnerability of distinct networks under different centrality methods.

where n is the number of variables, x_i and y_i are the single element with index i, $\bar{x}$ and $\bar{y}$ are the mean value.

Scatter plots with normalized centrality on horizontal and vertical axes between BPR1 and other typical measures on an ER network are shown with different colors in Fig. 3. For example, the first row represents the correlation between BPR measure and the other measures. It is found that none of the correlation coefficients between BPR measure and other measures exceed 0.5, suggesting that BPR behaves very



(a) (b) (c) (d) (e) (f)

FIGURE 5. Static attack on (a) ER network (b) SF network (c) European power grid (d) Oldenburg road network (e) C.elegans neural network (f) Protein-protein interaction network. The inset represents the complete figure of each network.

differently from other methods. The reason behind this is that BPR model aims to find the minimum feedback vertex set differing from the other method.

B. COMPARISON UNDER STATIC ATTACK STRATEGIES

Now we move on to study the vulnerability of a network by simultaneously removing a fraction of nodes. In the static scheme, the centrality measure of each node will not be recalculated when the structure of a network changes. First, we perform BPR and other benchmark methods on two different model networks and four real networks, which have been described in Section IV in detail. Here, we define a critical fraction of attack nodes q_c as the minimum fraction of removed nodes keeping the size of the largest connected component$\sigma \leq \sigma_c$. As seen in Fig 4a, BPR1 and BPR2 both have the lowest q_c on ER and SF network when $\sigma_c = 0.1$, while other methods to achieve the same effect need to remove a larger proportion of nodes. To further explore the

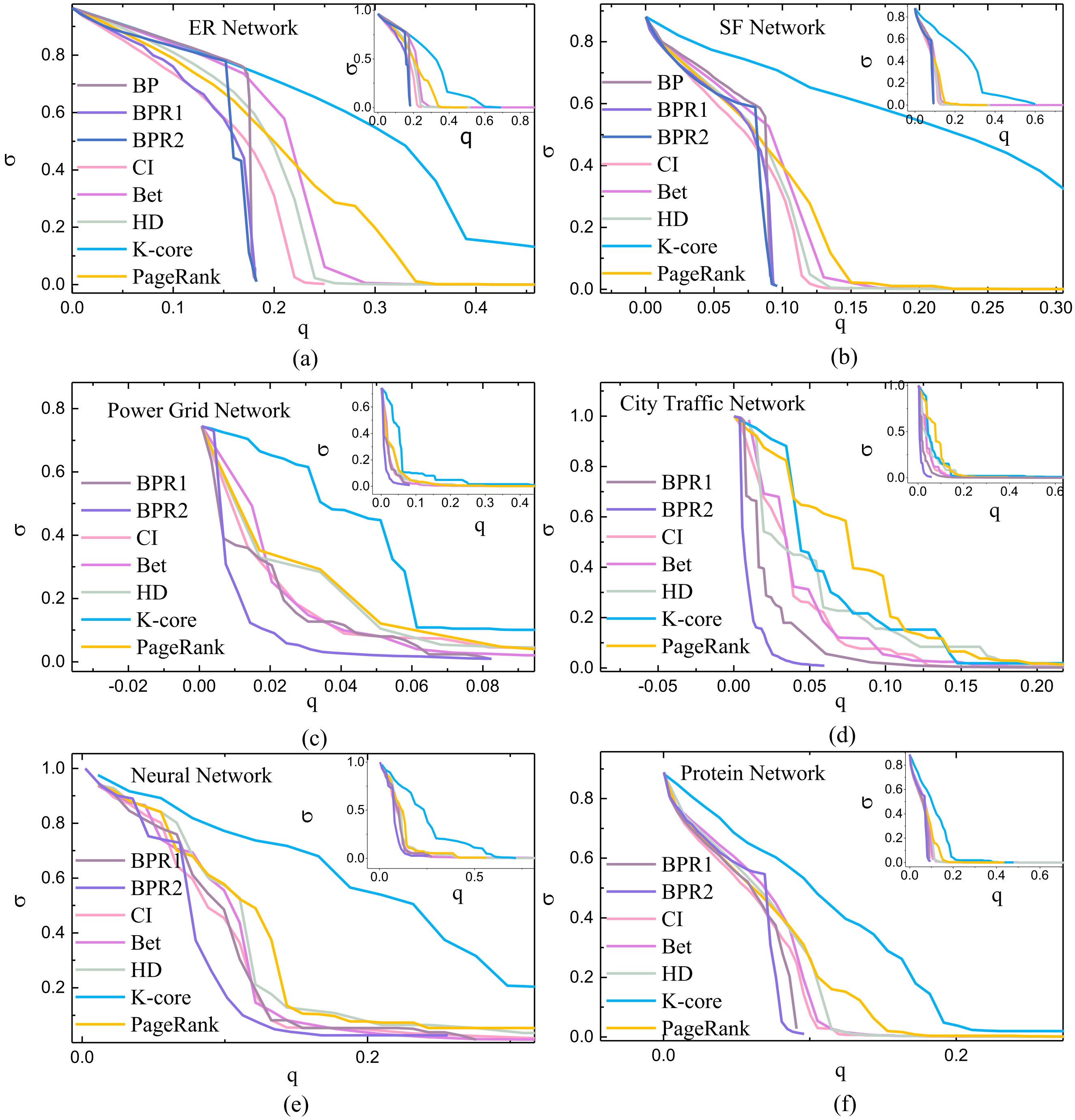


FIGURE 6. Dynamical attack on (a) ER network (b) SF network (c) European power grid (d) Oldenburg road network (e) C.elegans neural network (f) Protein-protein interaction network. The inset represents the complete figure of each network.

attack effects of different methods, we show the vulnerability V-index that has been defined before in Fig 4b. It is found that the vulnerability of networks can be clearly increased by BPR method compared with other methods.

In Fig. 5, the size of the largest connected component σ as a function of the fraction of removed nodes q is calculated for both model and real networks following distinct centrality measures, where 1 minus the area enclosed by the axis and curve represents V-index of a network. It is clear that our BPR measure with the smallest enclosed area is proved superior to other measures. Besides comparing with some typical methods, In Fig 5a and b, we also observe that BPR method offers a superior solution compared with recently proposed single BP [29], [30] and CI method [3], [41] at the initial stage ($0 < q < 0.1$), where σ by single BP and CI decrease much slower than that by BPR. This suggests that even though results in Fig 4 show BP and CI have a similar behavior with BPR, BPR can also help to identify critical nodes much more effectively at an early stage. We believe that the attack sequence mainly determines the attack effects during the failure process.

It is also found that network with a higher clustering coefficient exhibits a faster speed of structure collapse (see Fig. 5 and Table 3). We believe that network that appears to have a lower link density will be more robust than a higher one. Calculation of the ER network and SF network, whose degree distribution respectively follows a Poisson distribution and a power-law distribution, shows that SF network is more vulnerable according to these centrality measures. It is also found that the degree centrality that is a purely local measure is more effective than other measures that record global information of structure. Accordingly, degree centrality outperforms other measures at exposing network vulnerability in the static scheme whereas nodes identified by global measures like betweenness act as a bridge to connect highly connected parts of the structure.

Network topology highly affects the network robustness. It is known that ER network has a more robust structure than SF network under malicious attack. Experiments are also conducted on real networks of different fields in Fig 5c –f, where the attack strategy by BPR method is proved the most efficient at exposing the vulnerability of a network. We think that a network with the low average degree and long average path length will be easier to be degraded. The possible explanation for this may be a network with these properties is lack of spatially long-range correlation leading to a vulnerable network topology, which indicates that failure of the local structure may easily destroy the network.

C. COMPARISON UNDER DYNAMICAL ATTACK STRATEGIES

BPR can be used to identify vital nodes with an efficient performance in static attack scheme. The success of BPR leads us to the question: can we apply BPR method to a dynamical scheme? In the mode of dynamical attack, the importance of a node needs to be recalculated repeatedly after each round of attack. In other words, node centrality is updated iteratively during the whole attack process. Fig. 6 shows the size of the largest connected component σ under the sequential attack for networks and methods of different types. Compared with the static scheme, the dynamical scheme has a quite different behavior of destroying a network.

Network attacked by dynamical strategies appears to be more vulnerable (with a smaller enclosed area) than its static counterpart. In other words, dynamical attack scheme exhibits a greater attack effect than that of the static scheme due to the iterative calculation of importance of a node. However, not all the results follow this rule. Taking C.elegan neural network under K-core centrality measure for an example, the static attack is counter-intuitively more effective than a dynamical scheme. Moreover, different centrality measures show a different efficiency of network degrading, where the global measure BPR seems to be the most effective methods for both model and real networks compared with other measures.

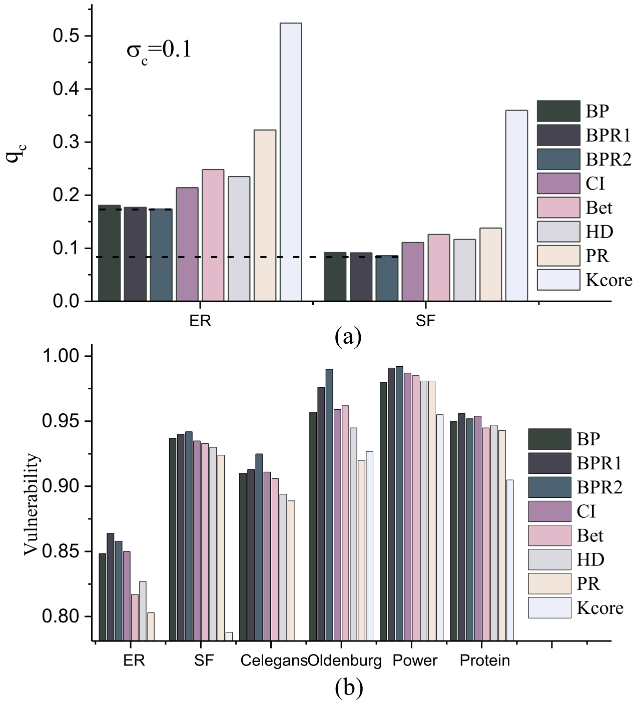


FIGURE 7. Dynamical attack. (a) Critical removal fraction q_C for methods of different types. (b) Vulnerability of distinct networks under different centrality methods.

In Fig 7, further study of critical removal fraction q_C and vulnerability V-index reveal that vital nodes of distinct networks can be located remarkably better and faster by BPR than by other methods. To quantitatively test our BPR method, we also calculate the performance improvement based on Eqs (12) between BPR and other methods. Results are shown in Table 4. It is found that BPR can be up to 4.35%∼76.11% more efficient than other methods in performance.

$$PI(i) = \frac{|V(BPR) - V(i)|}{V(i)}\% \tag{12}$$

where V stands for vulnerability metric after removing vital nodes, PI represents performance improvement, i can be different methods including BP, CI, Bet, HD, PageRank and K-core.

D. COMPUTATIONAL EFFICIENCY ANALYSIS

Here, we aim to analyze the complexity of BPR method and other methods. BPR consists of two parts: BP and node reinsertion, where BP is almost linear complexity with $O(N \ln N)$ and average complexity of searching, finding and deleting for node reinsertion is $O(\ln N)$. The "Union" operation of node reinsertion, merging the candidate node into a connected component based on the previous graph, has a complexity of $O(N)$. Therefore, the overall complexity of BPR can be

TABLE 4. Performance improvement on different networks.

i	BP	CI	Bet	HD	PageRank	Kcore
ER	5.29%	5.33%	22.40%	17.92%	51.70%	27.92%
SF	11.56%	10.77%	13.43%	17.14%	72.64%	23.68%
Neural	8.96%	15.73%	20.21%	29.25%	67.95%	32.43%
Road	62.78%	75.61%	73.68%	81.82%	86.30%	87.50%
Power	35.58	38.46%	46.67%	57.89%	82.22%	57.89%
Protein	5.89%	4.35%	12.73%	9.43%	49.47%	15.79%

TABLE 5. Time comparison on different networks using various methods (E and N respectively represents the number of edges and nodes).

BPR	BP	CI	Bet	HD	PageRank	Kcore
$O(N\ln N)$	$O(N\ln N)$	$O(N^2)$	$O(EN)$	$O(E+N)$	$O(E)$	$O(E)$
7E-4s	4E-4s	2.96s	30.67s	8E-3s	5E-4s	3E-4s

$O(N\ln N + k\ln N + N) \sim O(N\ln N)$. In Table 5, we show the complexity of different methods and their calculation time on an ER network with 10000 nodes and $<k> = 4$. (The results are calculated on an 8-core CPU computer with Intel i7 2.8 GHz and 16G memory and are averaged over 500 realizations). It is found that BPR has a relatively lower complexity than other methods.

VI. CONCLUSIONS

In summary, it is important to make sense of impacts on the integrity of the entire network against failure. In view of previous studies, many efforts have been devoted to study how the structure of a network changes after a fraction of vital nodes are removed according to different centrality measures. However, these methods perform differently in speed and accuracy when network topology varies, which may lead to underestimating the importance of some important nodes. In this paper, we examine the attack vulnerability of both model and real networks by vital nodes identification based on BPR method. Two steps are included in BPR algorithm to identify vital nodes and optimize attack order: BP process and node reinsertion. Moreover, two kinds of attack schemes are considered to test the efficiency of BPR in this paper: static attack strategy and dynamical attack strategy. On the one hand, we find that finding the minimum feedback vertex set for both attack schemes is more efficient than other global-based methods on different networks, indicating that nodes identified by BPR method should be targeted first. On the other hand, results demonstrate that BPR is able to achieve better performance in terms of speed, where the complexity of BPR is lower than other methods. However, due to the emergence property of vital nodes identified by BPR, the removal of a fraction of these vital nodes may lead to the abrupt breakdown of a network. This may make the BPR-guided strategy a dangerous scheme for destructive purposes. Moreover, BPR method may be only applicable to static single networks, where we need a further study to develop the new method for vital nodes identification in the temporal network and interdependent network. It is also possible for us to consider the impact of dynamical process for vital nodes identification, such as cascading failure and virus spreading.

There are many areas for application of BPR method such as city traffic, power grid, and even ecological system. For example, to best protect healthy people from infection, we need to immune a specific group of people within a limited time and resource, where identifying vital nodes at an early stage is a key. As an ocean of empirical networks can be characterized by multiple dependency links between networks whose functioning in one network highly depends on another one, the future work may be extended to investigate vulnerability of the interdependent network by vital nodes identification. Elucidation of the structural difference between the interdependent network and single network seems to be an important research due to its significance for network vulnerability and normal functioning.

REFERENCES

[1] M. Heo, S. Maslov, and W. Eaton, "Topology of protein interaction network shapes protein abundances and strengths of their functional and nonspecific interactions," *Proc. Nat. Acad. Sci. USA*, vol. 108, no. 10, pp. 4258–4263, 2011.

[2] P. J. Menck, J. Heitzig, J. Kurths, and S. H. Joachim, "How dead ends undermine power grid stability," *Nature Commun.*, vol. 5, p. 3969, Jun. 2014.

[3] F. Morone and H. A. Makse, "Influence maximization in complex networks through optimal percolation," *Nature*, vol. 527, no. 7579, pp. 65–68, 2015.

[4] R. R. Sharafat, C. L. Pu, R. B. Chen, and Z. Q. Xu, "Multiple-predators-based capture process on complex networks," *Chin. Phys. B*, vol. 26, no. 3, pp. 598–603, Mar. 2017.

[5] S. Bo, J. Guo-Ping, S. Yu-Rong, and X. Ling-Ling, "Rapid identifying high-influence nodes in complex networks," *Chin. Phys. B*, vol. 24, no. 10, pp. 1–9, Oct. 2015.

[6] *Blackout 2003: Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations*, U.S.-Canada Power System Outage Task Force, Washington, DC, USA, 2004. [Online]. Available: https://www.energy.gov/sites/prod/files/oeprod/DocumentsandMedia/BlackoutFinal-Web.pdf

[7] R. Cohen, K. Erez, D. Ben-Avraham, and S. Havlin, "Breakdown of the internet under intentional attack," *Phys. Rev. Lett.*, vol. 86, no. 16, pp. 3682–3685, 2001.

[8] Y. Chen, G. Paul, S. Havlin, F. Liljeros, and H. E. Stanley, "Finding a better immunization strategy," *Phys. Rev. Lett.*, vol. 101, no. 5, p. 058701, 2008.

[9] F. Altarelli, A. Braunstein, and L. Dall, "Asta, and R. Zecchina, "Optimizing spread dynamics on graphs by message passing," *J. Stat. Mech., Theory Experim.*, vol. 2013, no. 9, pp. 387–402, 2013.

[10] F. Altarelli, A. Braunstein, and L. Dall'Asta, J. R. Wakeling, and R. Zecchina, "Containing epidemic outbreaks by message-passing techniques," *Phys. Rev. X*, vol. 4, no. 2, p. 021024, 2014.

[11] L. Fei and Y. Deng, "A new method to identify influential nodes based on relative entropy," *Chaos Soliton Fractals*, vol. 104, pp. 257–267, Nov. 2017.

[12] D. Wang, H. Wang, and X. Zou, "Identifying key nodes in multilayer networks based on tensor decomposition," *Chaos*, vol. 27, no. 6, p. 063108, 2017.

[13] Z. Sun, B. Wang, J. Sheng, Y. Hu, Y. Wang, and J. Shao, "Identifying influential nodes in complex networks based on weighted formal concept analysis," *IEEE Access*, vol. 5, no. 99, pp. 3777–3789, 2017.

[14] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, "Network robustness and fragility: Percolation on random graphs," *Phys. Rev. Lett.*, vol. 85, no. 25, p. 5468, 2000.

[15] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, and T. Zhou, "Vital nodes identification in complex networks," *Phys. Rep.*, vol. 650, pp. 1–63, Sep. 2016.

[16] P. Bonacich, "Factoring and weighting approaches to status scores and clique identification," *J. Math. Social*, vol. 2, no. 1, pp. 113–120, 1972.

[17] M. Kitsak *et al.*, "Identification of influential spreaders in complex networks," *Nature Phys.*, vol. 6, pp. 888–893, Aug. 2010.

[18] M. A. Al-Garadi, K. D. Varathan, and S. D. Ravana, "Identification of influential spreaders in online social networks using interaction weighted K-core decomposition method," *Physica A, Stat. Mech. Appl.*, vol. 468, pp. 278–288, Feb. 2016.

[19] D. Chen, L. Lü, M.-S. Shang, Y.-C. Zhang, and T. Zhou, "Identifying influential nodes in complex networks," *Phys. A, Statist. Mech. Appl.*, vol. 391, pp. 1777–1787, Feb. 2012.

[20] R. Paluch, X. Lu, K. Suchecki, B. K. Szymański, and J. A. Hołyst, "Fast and accurate detection of spread source in large complex networks," *Sci. Rep.*, vol. 8, no. 1, p. 2508, 2018.

[21] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, Mar. 1977.

[22] L. C. Freeman, "Centrality in social networks conceptual clarification," *Soc. Netw.*, vol. 1, no. 3, pp. 215–239, 1979.

[23] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, vol. 18, no. 1, pp. 39–43, 1953.

[24] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 1–7, pp. 107–117, Apr. 1998.

[25] L. Lü, Y. C. Zhang, C. H. Yeung, and T. Zhou, "Leaders in social networks, the Delicious case," *PLoS ONE*, vol. 6, no. 6, p. e21202, 2011.

[26] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," presented at the ACM-SIAM Symp. Discrete Algorithms, 1998, pp. 668–677.

[27] V. Colizza, A. Flammini, M. A. Serrano, and A. Vespignani, "Detecting rich-club ordering in complex networks," *Nature Phys.*, vol. 2, no. 2, pp. 110–115, 2006.

[28] M. Mezard and G. Parisi, "The Be the lattice spin glass revisited," *Eur. Phys. J. B*, vol. 20, no. 2, pp. 217–233, 2001.

[29] S. Mugisha and H. J. Zhou, "Identifying optimal targets of network attack by belief propagation," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 94, no. 1, p. 012305, 2016.

[30] H. J. Zhou, "Spin glass approach to the feedback vertex set problem," *Eur. Phys. J. B*, vol. 86, no. 11, pp. 1–9, 2013.

[31] B. Wang, H. Tang, C. Guo, and Z. Xiu, "Entropy optimization of scale-free networks' robustness to random failures," *Physica A*, vol. 363, no. 2, pp. 591–596, 2005.

[32] V. Latora and M. Marchiori, "Efficient behavior of small-world networks," *Phys. Rev. Lett.*, vol. 87, p. 198701, Oct. 2001.

[33] A. Arulselvan, C. W. Commander, L. Elefteriadou, and P. M. Pardalos, "Detecting critical nodes in sparse graphs," *Comput. Oper. Res*, vol. 36, no. 7, pp. 2193–2200, 2009.

[34] C. M. Schneider, A. A. Moreira, A. J. Jr, S. Havlin, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proc. Nat. Acad. Sci. USA*, vol. 108, no. 10, pp. 3838–3841, 2011.

[35] P. Erdos and A. Rényi, "On the evolution of random graphs," *Pub. Math. Inst. Hungarian Acad. Sci.*, vol. 5, no. 1, pp. 17–60, 1960.

[36] G. Bianconi and A. L. Barabasi, "Competition and multiscaling in evolving networks," *Europhys. Lett.*, vol. 54, no. 4, pp. 436–442, May 2001.

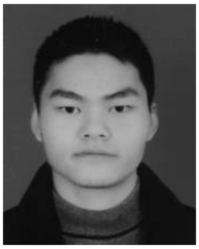
[37] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 72, p. 027104, Aug. 2005.

[38] F. Li. (2005). *Real Datasets for Spatial Databases: Road Networks and Points of Interest*. http://www.cs.utah.edu/~lifeifei/SpatialDataset.htm

[39] *ENTSOE*. (2015). https://www.entsoe.eu/data/Pages/default.aspx

[40] J. F. Rual *et al.*, "Towards a proteome-scale map of the human protein–protein interaction network," *Nature*, vol. 437, no. 7062, pp. 1173–1178, 2005.

[41] F. Morone, B. Min, B. Lin, R. Mari, and H. A. Makse, "Collective Influence Algorithm to find influencers via optimal percolation in massively large social media," *Sci. Rep*, vol. 6, p. 30062, Jul. 2016.

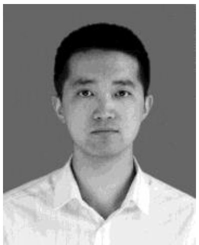


JILONG ZHONG is currently pursuing the Ph.D. degree with the Equipment Management and Safety Engineering College, Air Force Engineering University, Xi'an, China, and the School of Reliability and Systems Engineering, Beihang University, Beijing, China.

His current research interests include complex networks, cyber-physical system reliability modeling, and failure propagation.

FENGMING ZHANG is currently a Professor with Air Force Engineering University.

His research interests include complex systems, cyber-physical system reliability modeling, machine learning, and artificial intelligence.



ZHENGXIN LI received the Ph.D. degree from Air Force Engineering University, China, in 2011. He is currently a Lecturer with Air Force Engineering University.

His research interests include sequence pattern recognition, data mining, machine learning, and artificial intelligence.