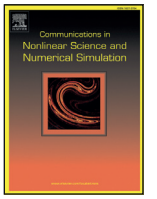
Commun Nonlinear Sci Numer Simulat 98 (2021) 105780

Contents lists available at ScienceDirect

Commun Nonlinear Sci Numer Simulat

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



Research paper

A Framework of Hierarchical Attacks to Network Controllability

Yang Lou[a], Lin Wang[b,c,*], Guanrong Chen[a]

[a] Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China
[b] Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
[c] Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 19 November 2020
Revised 17 January 2021
Accepted 18 February 2021
Available online 20 February 2021

Keywords:
Complex network
Attack strategy
Network controllability
Robustness

ABSTRACT

Network controllability robustness reflects how well a networked dynamical system can maintain its controllability against destructive attacks. This paper investigates the network controllability robustness from the perspective of a malicious attack. A framework of hierarchical attack is proposed, by means of edge- or node-removal attacks. Edges (or nodes) in a target network are classified hierarchically into categories, with different priorities to attack. The category of critical edges (or nodes) has the highest priority to be selected for attack. Extensive experiments on nine synthetic networks and nine real-world networks show the effectiveness of the proposed hierarchical attack strategies for destructing the network controllability. From the protection point of view, this study suggests that the critical edges and nodes should be hidden from the attackers. This finding helps better understand the network controllability and better design robust networks.

1. Introduction

Many real-world systems can be modeled as complex networks, which have gained growing recognition and popularity since the late 1990s, now becoming a self-contained discipline encompassing computer science, systems engineering, statistical physics, applied mathematics, and social sciences [1–4]. In practical applications, it is essential to determine whether or not a networked system can be controlled for utilization. Consequently, network controllability has become a focal research topic in network studies [5–16]. Same as the classical concept for systems, *controllability* here refers to the ability of a dynamical network being steered by external inputs from any initial state to any desired target state under an admissible control input within a finite duration of time.

On the other hand, random failures and malicious attacks on complex networks have become more and more frequent and severe recently [17–22]. Such failures and attacks take place in the form of node- and edge-removals, causing significant consequences to the systems such as malfunctioning or even completely crashing. For example, failures of traffic lights may cause traffic congestion in the urban transportation networks; neurological disorders may cause dysfunction or illness to humans. To resist attacks or failures, strong robustness is desirable and often necessary for a practical networked system. In different scenarios, there are different definitions and measures for network robustness [23]. Since theoretical analysis seems impossible for large-scale real-world networks, at least in the present time, the correlation between network robustness and

* Corresponding author.
E-mail addresses: felix.lou@my.cityu.edu.hk (Y. Lou), wanglin@sjtu.edu.cn (L. Wang), eegchen@cityu.edu.hk (G. Chen).

https://doi.org/10.1016/j.cnsns.2021.105780
1007-5704/

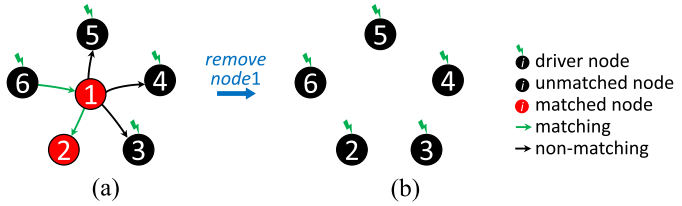


Fig. 1. [color online] Example of controllability and connectedness robustness: (a) given a star-shaped network, the number of required driver nodes is 4; its LCC is 6; (b) after the central hub is removed, the number of required driver nodes becomes 5; its LCC drastically dropped to 1.

topological features are generally investigated empirically, taking advantages of super-computing power available today [23–25]. In this pursuit, it is worth mentioning that the development of deep learning techniques offers an efficient option for empirical studies [26–28].

The notion of random failures and malicious attacks on complex networks, as well as the corresponding network robustness, covers a broad range of subjects. This paper concerns with the network *controllability robustness*, which refers to how well a networked dynamical system can maintain its controllability against random failures or, in particular, intentional attacks.

The issue of network robustness within different contexts regarding network topologies has been extensively investigated, and many edge- and node-removal attack strategies have been proposed to destruct the *connectedness* of the networks. Generally, attack strategies can be categorized into *random* and *targeted* attacks. A targeted attack aims at removing an intentionally selected object (e.g., the highest-degree node or the largest-betweenness edge), while a random attack do the removal randomly.

In the above studies, it is commonly assumed that necessary knowledge of the network is known and is recalculated after each attack, since it is reported that recalculated attacks are more destructive than the non-recalculated attacks [29]. For targeted attacks, it is also assumed that the targeted object is more important than the others in maintaining the network connectedness. Commonly used measures of importance include degree centrality, betweenness centrality, neighborhood similarity [30], branch weighting [31], and structural holes [32]. However, ranking the importance of nodes or edges is practically intractable for large-scale networks, since most measures cannot guarantee that removing the targeted object will definitely cause a greatest effect of damage on the network.

The size of the largest connected component (LCC) is widely used as a measure for connectedness robustness [19], which is the number of nodes in the largest weakly connected part of the network. A directed graph is *weakly connected* if all its directed edges are replaced by undirected edges, and the resultant undirected network is a connected one. It is observed that betweenness-based attacks may become less effective in the later stage of an attack process. This observation consequently leads to the effective conditional attack strategy: to remove the global highest-betweenness node only if it belongs to LCC; otherwise, to remove the local highest-betweenness node inside the LCC [33]. In [29], degree and betweenness are used simultaneously, with predefined weights for their balance, as the measure of importance. The module-based attack strategy [34,35] aims at attacking the nodes with inter-community edges, which are crucial to maintain the connectedness among communities. The damage-based attack [36] uses the degree of *damage* to measure the effectiveness of an attack, where the damage of an attack is defined by the change of the LCC size before and after the attack. It is also observed that the evolution process of attack and defense can enhance the network robustness [37], which is similar to the process of a mutual improvement of spears and shields.

Although the robustness of connectedness has a certain positive correlation with the robustness of controllability on a network, they actually have very different measures and objectives, as illustrated by the simple example shown in Fig. 1, where the *driver node* is a node to be controlled by an input so as to make the whole network become (or return to be) controllable (after the attack). This paper is concerned with the interplay of the connectedness, attack strategies, and controllability robustness of general complex networks.

Specifically, this paper focuses on the attack strategies that aim at destructing the network controllability. It is observed that removing highest-degree nodes [38] or highly-loaded edges [39] are more effective to degrade the network controllability than random removals. Furthermore, in [40] it is shown that node-removals are more harmful than edge-removals to the network controllability, and that heterogeneous networks are more vulnerable than homogeneous networks. Also, it is found that for many real-world networks the betweenness-based attacks are the most destructive to the controllability [40]. Moreover, it is reported in [41] that degree-based node-removal attacks cause greater damage to local-world networks [42] with a larger local-world size, while networks with smaller local-world sizes are more robust regarding both connectedness and controllability. A local-world network consists of multiple local communities; and local-world size refers to the number of nodes in a local community. Notably, the hierarchical structure of a directed network enables the random upstream (or downstream) attack, which removes the upstream (or downstream) node of a randomly-picked one, resulting in a more destructive attack strategy than the simple random attacks [20]. For a directed network, edges can be categorized into three types [5]: 1) *critical* edges, whose removal destroys the network controllability; 2) *redundant* edges, whose removal has no influence on the controllability; and 3) *ordinary* edges, whose removal will not change the number of needed driver

nodes, but may change the set of driver nodes. The critical edge attack strategy [43] collects all the critical edges from the initial network and remove them. After all the initial critical edges are removed, a random edge attack is performed. This is more destructive than a simple degree- or betweenness-based attack in the early stage of the process.

For the ultimate goal of protecting the network controllability and enhancing the controllability robustness, one can also study how a network can be effectively destructed. In this paper, a hierarchical framework is proposed for both node- and edge-removal attacks, aiming at maximizing the destruction of the network controllability. The main contributions of this work are: 1) the concept of *critical node* is introduced, quantified and analyzed, as a complement to the concept of critical edge; 2) a new hierarchical attack framework is proposed, which sorts the destruction of nodes or edges in a descending order, and is updated iteratively; 3) extensive simulations are performed to verify the effectiveness of the proposed methods, revealing that the exposure of critical edges and nodes is harmful to maintain a good network controllability.

The rest of the paper is organized as follows. Section 2 reviews the network controllability and its robustness, and several existing attack strategies. Section 3 introduces a new hierarchical attack framework. Section 4 evaluates both the hierarchical node- and edge-removal strategies by extensive numerical simulations, on both synthetic and real-world networks. Section 5 concludes the investigation.

2. Preliminary

2.1. Controllability and Controllability Robustness

A linear time-invariant (LTI) networked system [44], described by $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$, is *state controllable* if and only if the controllability matrix $[B\ AB\ A^2B\ \cdots A^{N-1}B]$ has a full row-rank, where A and B are constant matrices of compatible dimensions, $\mathbf{x}$ is the state vector, $\mathbf{u}$ is the control input, and N is the dimension of A. The *structural controllability* is its slight generalization dealing with two parameterized matrices A and B, in which the parameters characterize the structure of the underlying networked system: if there are specific parameter values that can ensure the system to be state controllable, then the system is structurally controllable. If the system is state controllable, its state vector $\mathbf{x}$ can be driven from any initial state to any target state in the state space within finite time by a suitable control input $\mathbf{u}$. Clearly, without control input $\mathbf{u}$, or $B \equiv 0$, the networked system is by no means controllable. Likewise, for a network of one-dimensional (scalar) nodes, there exist control inputs to some nodes to ensure its controllability. This network controllability is characterized by the minimum number of nodes with control inputs, called driver nodes, needed to maintain the controllability. When the network is put into the above LTI system formulation, how many and which nodes should be driver nodes are described by the matrix B.

Specifically, the controllability of a network of N scalar nodes is measured by the density of the driver nodes n_D, defined by

$$n_D \equiv \frac{N_D}{N}, \tag{1}$$

where N_D is the minimum number of driver nodes needed to retain the network controllability. Smaller n_D value represents better controllability. Practically, N_D can be calculated in two ways, for structural controllability and for exact (state) controllability, respectively. It was shown in [5] that identifying the minimum number of driver nodes to achieve a full control of a directed network requires searching for a maximum matching [45] of the network, which quantifies the network structural controllability. A matching is a set of edges that do not share start or end nodes; while a maximum matchingis a matching that contains the largest possible number of edges, which cannot be further expanded. A matched node means it is the end of a matching edge. When a maximum matching S^E is found and the set of matched nodes is denoted by S^N, where the superscripts E and N represent edge and node respectively, N_D is determined by the number of unmatched nodes, i.e., nodes without control inputs, given by

$$N_D = \max\{1, N - |S^N|\}, \tag{2}$$

where $|S^N|$ is the size of S^N, with $|S^N| \equiv |S^E|$, where $|S^E|$ is the size of S^E. As for exact controllability [6], N_D is calculated by

$$N_D = \max\{1, N - \mathrm{rank}(A)\}. \tag{3}$$

The controllability robustness is evaluated after some nodes or edges are removed, one by one, yielding a sequence of values (represented by a *controllability curve*) that reflect how robust (or vulnerable) a networked system is against a destructive attack. The controllability curve under a node-removal attack is calculated by

$$n_D^N(i) \equiv \frac{N_D(i)}{N - i}, \quad i = 0, 1, \ldots, N-1, \tag{4}$$

where $N_D(i)$ is the number of driver nodes needed to retain the network controllability after i nodes have been removed, and N represents the number of nodes in the original network. Note that, given an N-node network, one can remove at most $N-1$ nodes, excluding the trivial empty case. Similarly, the controllability curve under an edge-removal attack is calculated by

$$n_D^E(i) \equiv \frac{N_D(i)}{N}, \quad i = 0, 1, \ldots, M, \tag{5}$$

where $N_D(i)$ is the number of driver nodes needed to retain the network controllability after i edges have been removed, and N and M represent the numbers of nodes and edges in the original network. Here, $n_D^N(0) = n_D^E(0)$ represents the controllability of the original network, of which no node or edge has been removed.

The mean integrated percolation [46] is used to measure the overall controllability robustness, which averages the density of driver nodes after one node/edge is removed from the network, or equivalently it averages the convergence curves, denoted by R_c^N and R_c^E, respectively.

$$R_c^N = \frac{1}{N}\sum_{i=0}^{N-1} n_D^N(i), \tag{6}$$

and

$$R_c^E = \frac{1}{M+1}\sum_{i=0}^{M} n_D^E(i). \tag{7}$$

Lower R_c^N and R_c^E values mean better overall controllability robustness against node- and edge-removal attacks, respectively.

2.2. Network Centrality

Given an unweighted directed network represented by its adjacency matrix A, where each element a_{ij} is 1 if there exists an edge from node i to j; otherwise $a_{ij} = 0$. Its in- and out-degrees k_i^{in} and k_i^{out} [3] can be calculated as follows:

$$\begin{cases} k_i^{in} = \sum_{j=1}^{N} a_{ji} \\ k_i^{out} = \sum_{j=1}^{N} a_{ij} \end{cases} \tag{8}$$

where N is the number of nodes in the network.

Betweenness of node i [2] is defined as follows:

$$b_i = \sum_{i \neq s \neq t} \frac{\sigma_{st}(i)}{\sigma_{st}} \tag{9}$$

where σ_{st} represents the number of shortest paths from node s to t; $\sigma_{st}(i)$ represents the number of paths that pass through node i. If there exist a path from node s to t, then node t is *reachable* from s. Betweenness of edge (i, j) can be similarly defined by Eq. (9) with $\sigma_{ij}(i)$ being replaced by $\sigma_{st}(i, j)$ representing the number of paths that pass through edge (i, j).

Closeness [2,47] is defined as follows:

$$b_i = \left(\frac{\psi_i}{N-1}\right)^2 \cdot \sum_{j \neq i} \frac{1}{d_{ij}} \tag{10}$$

where ψ_i represents the number of reachable nodes from node i; d_{ij} is the distance from node i to its reachable node j.

2.3. Existing Attack Strategies

The most frequently used measures of importance are the *degree* and *betweenness* centralities. A weighted measure is given by

$$p_i = \alpha \times \frac{k_i}{\sum_{i=1}^{N} k_i} + \beta \times \frac{b_i}{\sum_{i=1}^{N} b_i}, \tag{11}$$

where k_i and b_i represent the degree and the betweenness of node i, p_i represents the probability of removing it, and α and β are weights, which are set manually in [29] with β being replaced by $1 - \alpha$ in [48]. Similarly, in [49] three parameters, α, β and γ, are used to control the weights of degree, betweenness and harmonic closeness, respectively. These measures have been used in the strategies to attack interdependent networks [48–52], networks of networks [53,54], and weighted networks [55].

The edge-removal attack strategy proposed in [43] aims at removing the critical edges of the initial network, and after all the initial critical (IC) edges have been removed, a random attack is performed. This IC attack strategy is specifically designed to degrade the network controllability, where the term 'critical' is defined for controllability. This attack is especially destructive in the early stage of the process, but becomes less effective in latter stages, because critical edges are changing during the process due to the removal of some other edges. An example is shown in Fig. 2 (a), where a non-critical edge (edge (2,4)) becomes critical after some edge-removals. Therefore, critical edges need to be updated throughout the attack process such that the damage to network controllability can be maximized.

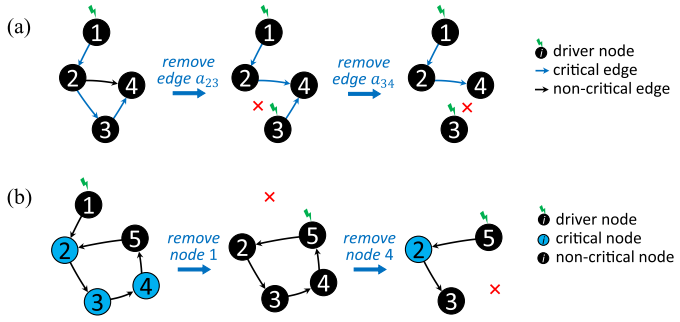


Fig. 2. [color online] An example of critical edges and nodes changing during the attack process: (a) edge (2,4) is non-critical in the initial network, but becomes critical after some edge-removals; (b) nodes 2,3 and 4 are critical initially, but after node 1 is removed, all of them become non-critical, and after node 4 is removed, node 2 becomes critical again.

2.4. Critical Edges and Nodes

In this paper, the concept of critical edge defined in [5] is adopted. An edge is critical if and only if its removal increases the number of driver nodes needed to retain the network controllability; otherwise, it is non-critical. Inspired by this, the concept of critical node is introduced here. A node is critical if and only if its removal increases the number of driver nodes needed to retain the network controllability; otherwise, it is non-critical. An example of critical node is shown in Fig. 2 (b), where the blue-colored nodes are critical nodes.

The critical nodes and edges are the most important elements in the concern of network controllability, in the sense that their removal will cause the maximum possible destruction to the network controllability. Therefore, in an efficient attack strategy, critical nodes and edges should be removed with the highest priority. It should be noted that, through the attack process, both critical nodes and edges will be dynamically changed, as illustrated by the example shown in Fig. 2. Therefore, in analyzing the attack strategy and its effect, the list of critical nodes and edges must be updated iteratively, step by step, after each attack.

3. Hierarchical Attack Framework

Since the removal of an edge or a node will increase the number of needed driver nodes by 1 at most, according to the definition, removing a critical edge/node will always cause a maximum damage to the network controllability. The main idea of the framework is to categorize the edges or nodes according to their priorities in attacks, the one with the highest priority one will always be removed first.

3.1. Hierarchical Edge Attack

The proposed framework classifies all edges hierarchically into three types: 1) critical edges, as defined above; 2) subcritical edges, whose removal does not increase the number of needed driver nodes, but increases the number of unmatched nodes; and 3) normal edges, which are the rest edges. The subcritical and normal edges are non-critical edges. In a hierarchical attack, the priorities of attacking these three types are in descending order, namely, selecting the critical edges with the highest priority to attack, followed by the subcritical ones, and finally the normal ones. Note that subcritical edges emerge only if there exists *one and only* perfect matching. To attack a subcritical edge is more harmful than to attack a normal edge, since the former breaks the perfect matching, while the latter does not. An example of these three types of edges is shown in Fig. 3.

Algorithm 1 shows the pseudo-code for hierarchical edge selection. Given a network with M edges, represented by its adjacency matrix A, Algorithm 1 returns the index of the edge to be removed with the highest priority. Lines 1–3 initialize three empty lists for the three types of edges. Lines 4 and 5 calculate the numbers of needed driver nodes and unmatched nodes of the original network before being attacked. The for-loop between Lines 6–20 categorizes each edge into a type list, namely, all the edges are categorized into the critical-list (L_1), the subcritical-list (L_2), or the normal-list (L_3), respectively. In Lines 21–27, the non-empty list with the highest priority is assigned to L, which is then sorted according to a certain feature F (e.g., degree centrality). Finally, $L(1)$ represents the index of an edge that is with the highest priority to be removed, and meanwhile it has the highest value of feature F (e.g., highest degree).

3.2. Hierarchical Node Attack

Nodes are hierarchically classified into three different types in descending order of priorities: 1) critical nodes; 2) normal nodes, whose removal does not affect the numbers of driver nodes; and 3) redundant nodes, whose removal enhances the controllability contrarily. The normal and redundant nodes are non-critical nodes. Note that there are no subcritical nodes,

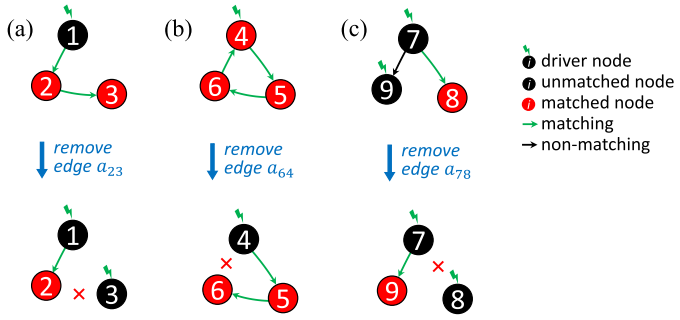


Fig. 3. [color online] Example of edge hierarchy: (a) edge (2,3) is critical, whose removal will increase the number of needed driver nodes by 1; (b) edge (4,6) is subcritical, whose removal will not change the number of driver nodes but will increase the number of unmatched nodes by 1; (c) edge (7,8) is normal, whose removal does not change the numbers of driver nodes and unmatched nodes.

Algorithm 1: Hierarchical Edge Selection

Input : adjacency matrix A; feature F; number of edges M; maximum matching S^E; size of maximum matching $|S^E|$
Output: index j of the edge to be attacked

1 $L_1 \leftarrow []$; `// highest priority`
2 $L_2 \leftarrow []$;
3 $L_3 \leftarrow []$; `// lowest priority`
4 $n_A \leftarrow$ *number of driver nodes needed for* A;
5 $u_A \leftarrow$ *number of unmatched nodes needed for* A;
6 **for** $i \leftarrow 1$ **to** $|S^E|$ **do**
7 $\quad A_0 \leftarrow A$;
8 $\quad$ *Delete edge* $S^E(i)$ *from* A_0;
9 $\quad n_{A_0} \leftarrow$ *number of driver nodes needed for* A_0;
10 $\quad u_{A_0} \leftarrow$ *number of unmatched nodes needed for* A_0;
11 $\quad$ **if** $n_{A_0} > n_A$ **then**
12 $\quad\quad L_1.insert(i)$;
13 $\quad$ **else**
14 $\quad\quad$ **if** $u_{A_0} > u_A$ **then**
15 $\quad\quad\quad L_2.insert(i)$;
16 $\quad\quad$ **else**
17 $\quad\quad\quad L_3.insert(i)$;
18 $\quad\quad$ **end**
19 $\quad$ **end**
20 **end**
21 **if** L_1 *is not empty* **then**
22 $\quad L \leftarrow L_1$;
23 **else if** L_1 *is empty* ***and*** L_2 *is not empty* **then**
24 $\quad L \leftarrow L_2$;
25 **else**
26 $\quad L \leftarrow L_3$;
27 **end**
28 *Sort* L *according to feature* F*, in descending order*;
29 $j \leftarrow L(1)$;

since if there is a unique perfect matching, all nodes are matched. Attacking any node will not change the number of driver nodes, but will break the perfect matching. An example of these three types of nodes is shown in Fig. 4.

Algorithm 2 shows the pseudo-code for hierarchical node selection. Given a network with N nodes, represented by its adjacency matrix A, Algorithm 2 returns the index of the node to be removed with the highest priority. Lines 1–3 initialize three empty lists for the three types of nodes. Line 4 calculates the numbers of needed driver nodes of the original network before being attacked. The for-loop between Lines 5–16 categorizes each node into a type list, namely, all the nodes are categorized into the critical-list (L_1), the normal-list (L_2), or the redundant-list (L_3), respectively. In Lines 17–23, the non-empty list with the highest priority is assigned to L, which is then sorted according to certain feature F. Finally, $L(1)$ represents the index of a node that has the highest priority to be removed, and meanwhile it has the highest value of feature F.

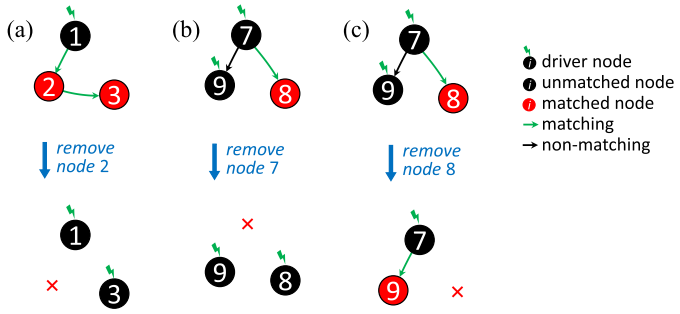


Fig. 4. [color online] Example of node hierarchy: (a) node 2 is critical, whose removal increases the number of needed driver nodes by 1; (b) node 7 is normal, whose removal does not change the numbers of driver nodes; (c) node 8 (or node 9) in the above network is redundant, whose removal decreases the number of needed driver nodes by 1.

Algorithm 2: Hierarchical Node Selection

Input : adjacency matrix A; feature F; number of nodes N; matched node set S^N; size of matched node set $|S^N|$
Output: index j of the node to be attacked

$L_1 \leftarrow []$; `// highest priority`
$L_2 \leftarrow []$;
$L_3 \leftarrow []$; `// lowest priority`
$n_A \leftarrow$ *number of driver nodes needed for* A;
for $i \leftarrow 1$ **to** $|S^N|$ **do**
  $A_0 \leftarrow A$;
  Delete node $S^N(i)$ *from* A_0;
  $n_{A_0} \leftarrow$ *number of driver nodes needed for* A_0;
  if $n_{A_0} > n_A$ **then**
    $L_1.insert(i)$;
  else if $n_{A_0} = n_A$ **then**
    $L_2.insert(i)$;
  else
    $L_3.insert(i)$;
  end
end
if $L1$ *is not empty* **then**
  $L \leftarrow L_1$;
else if $L1$ *is empty* ***and*** L_2 *is not empty* **then**
  $L \leftarrow L_2$;
else
  $L \leftarrow L_3$;
end
Sort L *according to feature* F*, in descending order*;
$j \leftarrow L(1)$;

Source codes of both hierarchical node and edge attack algorithms are available for the public[1].

3.3. Extra Computational Complexity

The computational complexity of calculating the network controllability, mainly in searching for the number of needed driver nodes, is $O(M \cdot \sqrt{N})$, by the Hopcroft–Karp algorithm. In a hierarchical node or edge attack, to identify whether a node or edge is critical, the number of needed driver nodes to be calculated iteratively introduces a non-negligible amount of extra computational cost.

Note that a critical edge or node must be a matching edge or a matched node, since the removal of an unmatched edge or node will not increase the number of needed driver nodes. The extra computational cost for hierarchical attack is

[1] https://fylou.github.io/sourcecode.html

Table 1
Basic information of the real-world networks.

Network	File name	Brief description	N	M
BMK	bn-mouse-kasthuri-graph-v4	brain network	1029	1559
ICM	ia-crime-moreno	interaction network	830	1474
IEU	inf-euroroad	infrastructure network	1175	1417
DEL	delaunay-n10	DIMACS10 problem	1024	3056
DW5	dwt-1005	symmetric connection from Washington	1005	3808
DW7	dwt-1007	symmetric connection from Washington	1007	3784
LSH	lshp1009	Alan Georges L-shape problem	1009	2928
OLM	olm1000	computational fluid dynamics problem	1000	2996
RAJ	rajat19	Rajat19 circuit simulation matrix	1157	4433

$O(\sqrt{N}\cdot|S^E|^2)$, where $|S^E|\equiv|S^N|\le N$ represents the number of edges in the maximum matching S^E and the number of nodes in the matched node set S^N, respectively.

In this paper, the attack performance is the primary consideration, while the computational cost could be high if the network size is large. Empirically, on a PC with 64-bit operation system and Intel i7-6700 (3.4 GHz) CPU, which is the computing hardware used in Section 4, the run time for a random node attack on a network of $N=1000$ is about 11.2 seconds, while it is 121.7 seconds for a hierarchical random attack. A possible way to reduce the computational burden can be considered as follows: Given a feature F, all the edges (or nodes) are sorted according to F in descending order first. Then, the criticalness of each edge (or node) is checked in the F-descending order. It stops when a critical edge (or node) is found. This offers an alternative way to executing Algorithms 1 and 2, with less computational burden. But this complicated issue is out of the scope of the present paper, which will be investigated in the future.

4. Experimental Studies

In this section, the hierarchical attack framework is evaluated by extensive simulations. Network features will be taken into account. For node attacks, betweenness, out-degree and closeness are used as feature F, respectively; for edge attacks, betweenness and degree are used, respectively. To verify the effectiveness of the proposed hierarchical framework, the hierarchical feature-based attack strategies are compared to the feature-based attack strategies, respectively. For example, the hierarchical degree-based attack is compared to the degree-based attack, under the same conditions.

Nine typical directed synthetic network models are adopted for simulation, namely the Erdös–Rényi random-graph (ER) network [56], Newman–Watts small-world (SW) network [57], generic scale-free (SF) network [38,58,59], q-snapback (QS) network [60], q-snapback network with redirected edges (QR) [61], random triangle (RT) network [25], and random rectangle (RR) network [25], extremely homogeneous (HO) network [62], and onion-like (OL) network [63].

Recall that the HO networks were empirically verified with optimal controllability robustness before [62]. Given an N-node and M-edge configuration, the HO network satisfies $\lfloor M/N\rfloor\le k_i^{in,out}\le\lceil M/N\rceil$, $i=1,2,\ldots,N$. This means that both of its in- and out-degrees are distributed identically or nearly identically with a small difference less than 1. The OL network is generated from an SF network via simple edge-swapping with degree reservation [63], thus its degree distribution follows the same power-law distribution as the SF.

The detailed generation methods and parameter settings of these synthetic networks are provided in Supplementary Information (SI)[2]. The network size is set to $N=500$, 1000, and 1500, respectively. The average degree is set to $\langle k\rangle=3$, 5, and 10, respectively.

In addition, nine real-world networks are used for simulations, with data taken from Network Repository[3]. Their parameters and brief descriptions are presented in Table 1.

4.1. Comparison of Attack Strategies

4.1.1. Node-removal attacks

Nine node-removal attack strategies are compared, namely the betweenness-based (N-B), out-degree-based (N-D), closeness-based (N-C), random (N-R), hierarchical betweenness-based (N-HB), hierarchical out-degree-based (N-HD), hierarchical closeness-based (N-HC), hierarchical random (N-HR), and hybrid (N-Hy) attacks.

The B, D, C strategies aim at removing the node with the largest betweenness, degree, and closeness, respectively, at every step. The HB, HD, HC strategies aim at removing critical nodes that are sorted in a betweenness-, degree-, and closeness-descending order, respectively, at every step. The HR removes the critical nodes at random.

The Hy strategy is designed as follows: First, remove either the node (or edge) with the maximum degree (or betweenness), according to the removal of which node (or edge) will cause greater destruction to the network controllability. If equal, then choose either one to attack.

[2] https://fylou.github.io/pdf/hatk_si.pdf
[3] http://networkrepository.com/

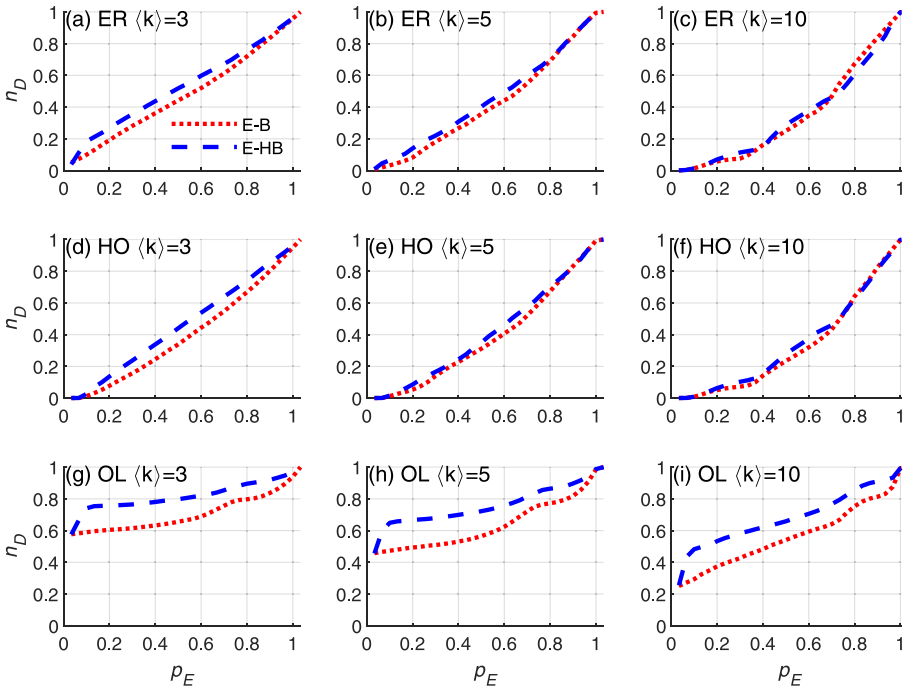


Fig. 5. [color online] Densities of driver nodes under edge attacks on ER, HO, and OL networks ($N = 1000$): hierarchical betweenness-based (E-HB) and betweenness-based (E-B) strategies.

4.1.2. Edge-removal attacks

Eight edge attack strategies are compared, namely the betweenness-based (E-B), out-degree-based (E-D), random (E-R), hierarchical betweenness-based (E-HB), hierarchical out-degree-based (E-HD), hierarchical random (E-HR), initial critical (E-IC) [43] and hybrid (E-Hy) attacks.

Here, as suggested in [43], the 'out-in' edge degree is used as the edge degree. For an edge $a_{i,j}$, its edge degree is calculated by $k_i^{out} + k_j^{in}$, i.e., the sum of the out-degree of its source node and the in-degree of its target node.

For each node and each edge attack, the simulation repeats 30 and 20 independent runs, respectively.

4.2. Simulation Results on Synthetic Networks

Here, the structural controllability (see Eq. (2)) is considered for controllability robustness comparison. The simulation results of some synthetic networks with $N = 1000$ and real-world networks are presented. More detailed and complete results for networks with $N = 500$, 1000 and 1500 are given in the SI.

Fig. 5 shows the results of ER, HO, and OL under E-HB and E-B attacks. It is clear that E-HB is consistently more destructive than E-B throughout the entire process as shown in Fig. 5 (a,b,d,e,f,g,h,i); while Fig. 5 (c) shows that E-HB is more destructive than E-B when $P_E < 0.7$, but E-B is slightly more destructive when $P_E > 0.7$.

Fig. 6 shows that E-HD is consistently more destructive than E-D. Fig. 7 shows that E-HR is consistently more destructive than E-R and E-IC. Figs 5–7 show that HO has better controllability robustness than ER; both HO and ER have significantly better controllability robustness than OL. As the average degree increases, the controllability robustness improves.

Fig. 8 compares the three hierarchical attacks (E-HB, E-HD and E-HR) and hybrid attack (E-Hy). For ER and HO, it is clear that E-HB is more destructive than the other strategies; while for OL, either E-HR or E-HB is most destructive. Figs. 5–8 show that the hierarchical framework increases the destruction effects on the network controllability. It can also be seen that, among the hierarchical attack strategies, E-HB performs the best.

The overall comparison is summarized in Table 2, where each value is the ratio of the overall destruction (see Eq. (6) for node attacks and Eq. (7) for edge attacks) under the two corresponding attack strategies. For example, the value 1.210 in row '$\langle k\rangle = 3$, ER' and column 'Node Attack, HB/B' represents that, given an ER ($\langle k\rangle = 3$) under node attacks, the overall destruction ratio of N-HB versus N-B is 1.210. Referring to Fig. 5 (a), the value is equivalent to the ratio of the area under the blue dashed-line versus the area under the red dotted-line.

If HB/B > 1, it means that HB is more destructive; if HB/B < 1, it means that B is more destructive; otherwise, if HB/H $= 1$, it means that HB and B are equivalently destructive. Here, an equivalent overall destruction does not mean that the two controllability curves are overlapped, but means that the areas under the two controllability curves are equal, namely they are equivalent in the average sense.

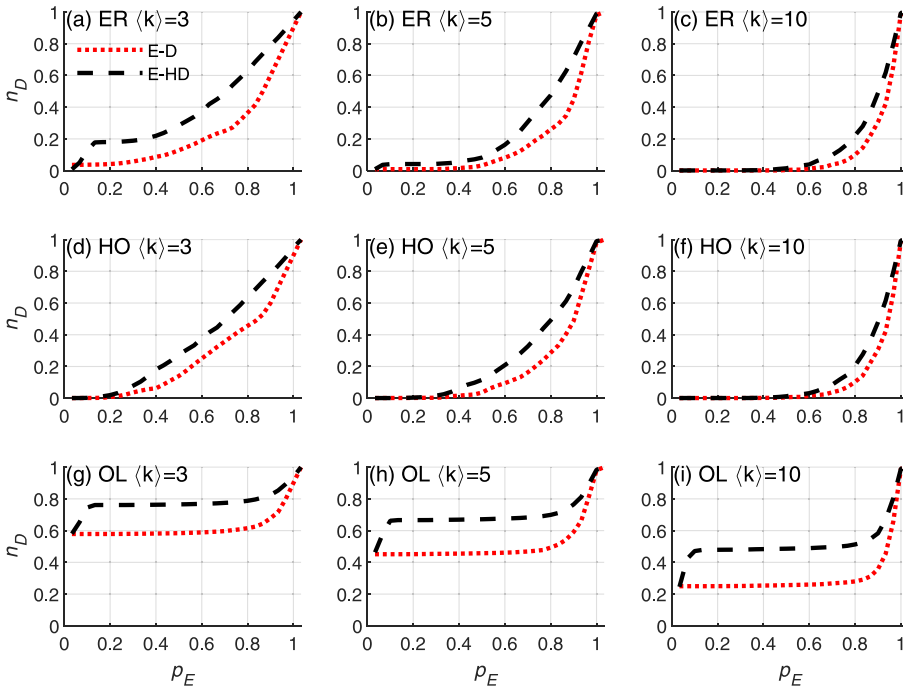


Fig. 6. [color online] Densities of driver nodes under edge attacks on ER, HO, and OL networks ($N = 1000$): hierarchical degree-based (E-HD) and degree-based (E-D) strategies.

As can be seen from the 'Node Attacks' part in Table 2, the hierarchical attack strategies are consistently more destructive than the non-hierarchical and the hybrid strategies, for the ratio values are greater than 1 in the columns of HB/B, HD/D, HC/C, and HR/R.

In the columns of HB/Hy and HD/Hy, hierarchical strategies generally outperform the hybrid strategy. It should be noted that, when the ratio is within [0.990,1.010], one may consider the two comparing strategies to have equivalent performances.

As for the edge-removal attacks, the hierarchical strategies are more destructive than the non-hierarchical ones and the IC (which does not update the critical edge list). However, the ratio values in the column of HD/Hy are mostly less than 1, meaning that HD is less destructive than Hy. This implies that edge degree is not a good measure of importance regarding destructive attacks. Nevertheless, within the hierarchical framework, E-HD is more destructive than E-D.

Overall, there are $304/324 = 93.8\%$ cases, showing the more destructive effectiveness of the hierarchical attacks.

The results for cases of $N = 500$ and $N = 1500$ are tabled in SI.

The attack simulation results on real-world networks are shown in Table 3, while the detailed comparison figures for different networks are included in SI.

There are 6 out of 108 values less than 1 in Table 3, all in the columns 'HD/Hy'. This implies that degree is a less destructive feature than betweenness for both edge and node attacks to these real-world networks. Nevertheless, there are $102/108 = 94.4\%$ cases that have verified the destructive performance of the hierarchical attacks to real-world networks.

The attack simulations on various synthetic networks and real-world networks show that the hierarchical strategies are consistently more destructive to network controllability than other attack strategies.

4.3. Critical Edges and Nodes

A common phenomenon is observed from the results presented in Sec. 4.2: as the average degree increases, the ratio of areas under the controllability curves subject to hierarchical and non-hierarchical attacks tends asymptotically to 1. This is because, as the network becomes denser, hence more homogeneous, fewer critical edges and nodes are exposed. It not only improves the controllability robustness of the network, but also makes the proposed hierarchical attack strategies less effective, thereby becoming similar to non-hierarchical attacks.

Table 4 shows the minimum (integer) average degree when there is no critical nodes or edges found in the initial network. Here, initial network means the network that has not been attacked. For each topology with a given $\langle k \rangle$ value, 30 network instances are simulated. Given $N = 500$, $\langle k \rangle$ is set from 3 with an incremental value 1. If there are no critical nodes or edges found in all the 30 instances, then the $\langle k \rangle$ value is recorded into Table 4; otherwise, $\langle k \rangle$ increases by 1 and then the process is run again. It can be seen from the table that, for SW and HO, there are no critical nodes or edges found when $\langle k \rangle = 3$, meaning that removal of any node or edge in the initial SW or HO will not increase the number of needed driver

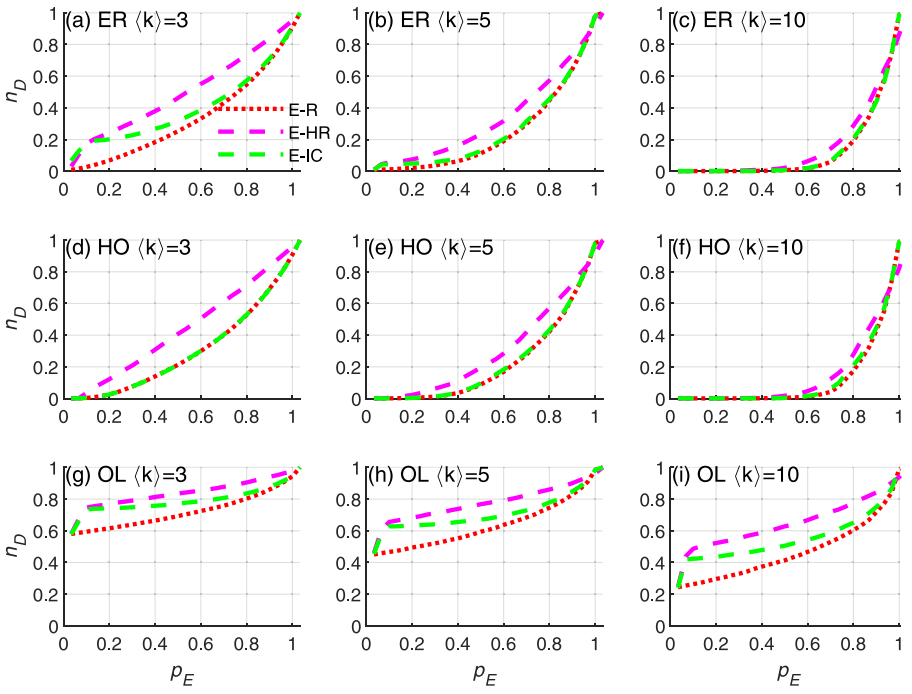


Fig. 7. [color online] Densities of driver nodes under edge attacks on ER, HO, and OL networks ($N = 1000$): random (E-R), hierarchical random (E-HR) and initial critical (IC) strategies.

nodes. Thus, their controllability robustness is better than the others. In contrast, for SF and OL, until $\langle k\rangle$ increases to 22 and 24, respectively, there are no critical nodes or edges. It means that in dense SF or OL networks (e.g., $\langle k\rangle = 20$), there are still critical nodes and edges, and removing any critical node or critical edge will directly destroy its controllability. Thus, SF and OL have much worse initial controllability and controllability robustness than the other networks.

Fig. 9 shows the number of critical edges in the initial SF and OL networks, against the increase of average degree. The corresponding figure for the case with critical nodes is shown in SI. The initial controllability is plotted for reference in each subplot. As shown in Fig. 9, the numbers of critical edges in the initial SF and OL networks increase as $\langle k\rangle$ increases from 3 to 13; when $\langle k\rangle > 16$, the numbers of critical edges drop drastically. Meanwhile, the initial controllability of both SF and OL becomes better as the average degree increases. When $3 \leq \langle k\rangle \leq 13$, the additional edges enhance the connectedness and make the networks more controllable. These additional edges become part of the critical edges. However, when $\langle k\rangle > 16$, the initial controllability of the SF networks tends to be sufficiently optimized, reflected by the lower density of needed driver nodes. In this case, the increased edges cover the critical nodes and edges, which leads to the drastically drops of the numbers of (the exposed) critical edges.

The exposure of critical nodes and edges sets a clear target for the attacker to destroy the network controllability. In contrast, in the networks with strong controllability robustness, there are rare (or no) critical nodes and edges exposed; for example, SW, HO, QS and QR. For these networks, the attacker is unable or uneasy to find targets to attack in order to destruct the controllability. This finding is consistent with, and actually extends the applicability of, the previous findings: 1) dense and homogeneous networks have better controllability [5]; 2) extremely-homogeneous topology has the optimal controllability robustness [62]. Nevertheless, critical nodes and edges will expose themselves during the attack process, as the network becomes sparser. To design networks with good controllability robustness, the exposure of critical nodes and edges should be dimmed or avoided, if ever possible. If there are sufficient numbers of available edges, the networks should be designed as dense and homogeneous as possible [5,62]; otherwise, if the numbers of edges are limited, they should be deliberately assigned in such a way that the exposure of critical nodes and edges is minimum.

Fig. 10 shows the types of removed nodes and edges during a single attack simulation. A green square means the increment of extra driver nodes; a black square represents a reduction of the driver nodes; a blue square means there is a perfecting matching in the current network; and a red square means nothing is changed to the network controllability.

As can be seen from Fig. 10 (a), HO does not expose critical nodes in the early stage of an N-HB attack. In Fig. 10 (b), HO and SW do not expose critical nodes in the early stage of an N-HR attack. It is also notable that, until the end of N-HR attacks, HO does not show redundant nodes. It is clear that network controllability is more destructively destroyed under N-HB attacks and more redundant nodes emerged. Comparing Figs. 10 (c) and (d), E-HB is more destructive in the early stage, but it exhausts critical edges in the latter stage. E-HR is less destructive in the early stage. In addition, E-HB can destroy the perfect matching rapidly, while E-HR cannot, reflected by the fact that there are consecutive blue squares in

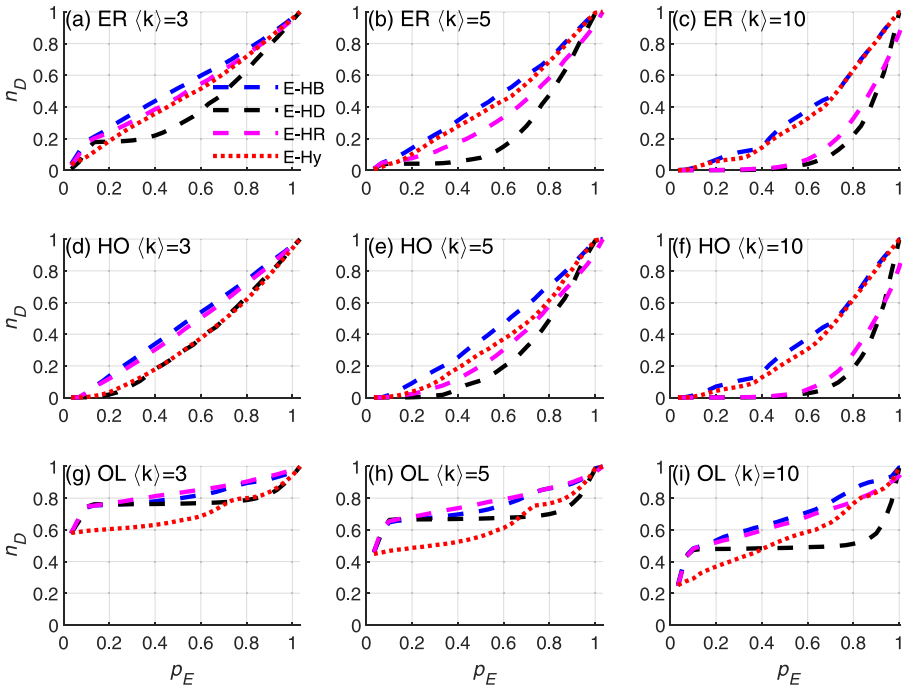


Fig. 8. [color online] Densities of driver nodes under edge attacks on ER, HO, and OL networks ($N = 1000$): three hierarchical attacks (E-HB, E-HD and E-HR) and hybrid (E-Hy) strategies.

Table 2
Comparison of attack strategies on the nine synthetic networks ($N = 1000$), where B represents betweenness; D represents degree; C represents closeness; R represents random; Hy represents hybrid; IC represents initial critical edges; HB represents hierarchical betweenness; HD represents hierarchical degree; HC represents hierarchical closeness; HR represents hierarchical random attacks.

N=1000		Node Attack (N-)						Edge Attack (E-)					
		HB/B	HD/D	HC/C	HR/R	HB/Hy	HD/Hy	HB/B	HD/D	HR/R	HB/Hy	HD/Hy	HR/IC
$\langle k\rangle = 3$	ER	1.210	1.151	1.313	1.469	1.111	1.132	1.112	1.585	1.454	1.123	**0.848**	1.225
	SW	1.286	1.070	1.209	1.275	1.065	1.081	1.157	1.339	1.426	1.295	**0.997**	1.426
	SF	1.033	1.011	1.029	1.266	1.007	1.009	1.158	1.246	1.157	1.157	1.101	1.060
	QS	1.303	1.198	1.661	1.344	1.098	1.185	1.123	2.256	1.463	1.123	**0.886**	1.241
	QR	1.336	1.160	1.330	1.445	1.138	1.155	1.118	1.721	1.440	1.122	**0.874**	1.411
	RT	1.148	1.095	1.164	1.590	1.056	1.070	1.260	1.679	1.393	1.280	1.072	1.202
	RR	1.245	1.110	1.240	1.527	1.095	1.098	1.217	1.682	1.438	1.231	1.013	1.282
	HO	1.247	1.170	1.436	1.416	1.141	1.148	1.151	1.336	1.378	1.095	1.143	1.382
	OL	1.034	1.012	1.028	1.273	1.007	1.007	1.161	1.241	1.161	1.164	1.104	1.060
$\langle k\rangle = 5$	ER	1.207	1.209	1.339	1.483	1.149	1.165	1.051	1.377	1.354	1.052	**0.627**	1.272
	SW	1.296	1.171	1.337	1.383	1.135	1.150	1.067	1.421	1.261	1.142	**0.756**	1.292
	SF	1.051	1.017	1.048	1.371	1.015	1.015	1.179	1.348	1.208	1.191	1.101	1.084
	QS	1.235	1.268	1.984	1.348	1.167	1.214	1.066	2.097	1.476	1.054	**0.732**	1.477
	QR	1.273	1.197	1.345	1.502	1.152	1.156	1.078	1.416	1.366	1.077	**0.673**	1.370
	RT	1.201	1.152	1.222	1.630	1.110	1.118	1.106	1.543	1.378	1.176	**0.751**	1.339
	RR	1.239	1.154	1.264	1.545	1.109	1.120	1.111	1.435	1.305	1.132	**0.756**	1.326
	HO	1.234	1.176	1.399	1.431	1.150	1.136	1.074	1.413	1.258	1.026	**0.994**	1.246
	OL	1.048	1.019	1.046	1.367	1.010	1.010	1.205	1.340	1.199	1.199	1.098	1.077
$\langle k\rangle = 10$	ER	1.214	1.208	1.320	1.450	1.148	1.117	1.025	1.325	1.202	1.030	**0.470**	1.221
	SW	1.217	1.208	1.363	1.376	1.149	1.113	1.064	1.238	1.127	1.102	**0.479**	1.164
	SF	1.078	1.036	1.080	1.590	1.025	1.027	1.205	1.608	1.336	1.197	**0.916**	1.155
	QS	1.167	1.315	2.875	1.295	1.198	1.195	1.036	1.777	1.395	1.188	**0.576**	1.369
	QR	1.228	1.203	1.359	1.427	1.148	1.116	1.029	1.341	1.154	1.044	**0.487**	1.137
	RT	1.211	1.208	1.286	1.496	1.121	1.128	1.075	1.332	1.280	1.057	**0.515**	1.221
	RR	1.224	1.169	1.300	1.409	1.106	1.119	1.090	1.309	1.214	1.083	**0.555**	1.187
	HO	1.201	1.194	1.361	1.376	1.157	1.140	1.034	1.275	1.063	1.078	**0.697**	1.063
	OL	1.066	1.034	1.079	1.577	1.021	1.022	1.189	1.619	1.369	1.212	**0.932**	1.167

Table 3
Comparison of attack strategies on the nine real-world networks, where B represents betweenness; D represents degree; C represents closeness; R represents random; Hy represents hybrid; IC represents initial critical edges; HB represents hierarchical betweenness; HD represents hierarchical degree; HC represents hierarchical closeness; HR represents hierarchical random attacks.

	Node Attack (N-)						Edge Attack (E-)					
	HB/B	HD/D	HC/C	HR/R	HB/Hy	HD/Hy	HB/B	HD/D	HR/R	HB/Hy	HD/Hy	HR/IC
BMK	1.066	1.007	1.010	1.157	1.004	1.005	1.031	1.063	1.056	1.030	**0.988**	1.027
ICM	1.044	1.102	1.183	1.361	1.033	1.041	1.101	1.201	1.187	1.100	1.004	1.088
IEU	1.187	1.050	1.137	1.361	1.007	1.040	1.109	1.377	1.291	1.115	1.097	1.096
DEL	1.203	1.174	1.191	1.311	1.064	1.057	1.108	1.416	1.248	1.112	**0.889**	1.163
DW5	1.276	1.149	1.456	1.423	1.097	**0.995**	1.167	1.483	1.381	1.169	**0.838**	1.237
DW7	1.323	1.036	1.597	1.314	1.004	**0.994**	1.205	2.224	1.284	1.235	1.291	1.269
LSH	1.301	1.312	1.977	1.264	1.066	1.086	1.121	1.585	1.301	1.122	**0.907**	1.217
OLM	1.007	1.001	1.009	1.946	1.010	1.019	1.281	1.660	1.611	1.696	1.627	1.649
RAJ	1.297	1.076	1.252	1.769	1.064	1.082	1.508	1.749	1.502	1.805	1.403	1.292

Table 4
The lowest average out-degree when there is no critical nodes or edges found in the network.

	ER	SW	SF	QS	QR	RT	RR	HO	OL
Node	9	3	23	6	4	8	6	3	22
Edge	10	3	24	5	6	8	6	3	22

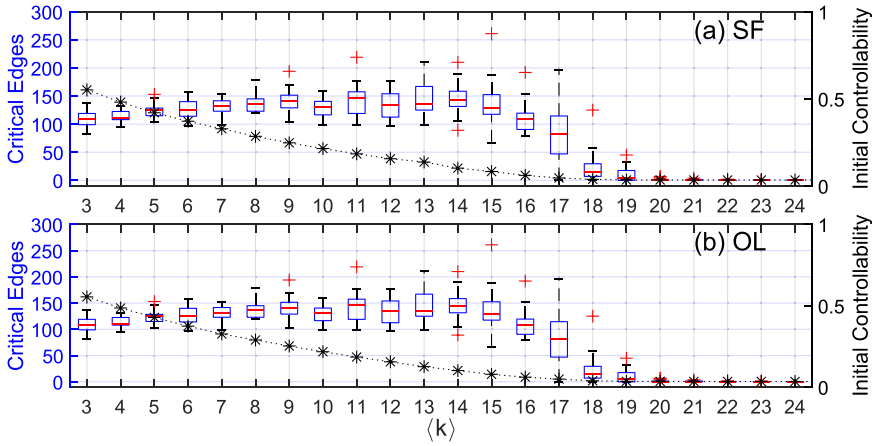


Fig. 9. Number of critical edges (boxplots) and initial controllability (stars *) against the average degree of (a) SF and (b) OL networks.

Fig. 10 (d), but not in Fig. 10 (c). In Figs. 10 (b) and (d), HO and SW do not expose critical edges in the early stage due to their homogeneity.

5. Conclusions

To better understand the network controllability robustness from the perspective of destructive attacks, a hierarchical attack framework is proposed, which can be used for both edge- and node-removal attacks. The hierarchical attack strategies aim at removing the critical nodes and critical edges with the highest priority, and they can be combined together with other commonly used features (e.g., degree centrality), such that the identified critical nodes or critical edges can be sorted in descending order according to such features. Extensive experiments on nine synthetic networks with various configurations and nine real-world networks show the effectiveness of the proposed hierarchical attack framework on destructive attacks to network controllability for all kinds of networks that are tested. For node attacks, betweenness, out-degree, and closeness are used as the feature, respectively; for edge attacks, betweenness and degree are used, respectively. The hierarchical feature-related attacks show consistently better destructive performances than the common feature-only attacks.

It is revealed that the exposure of the critical edges and nodes are disadvantageous in resisting attacks to the network controllability. Therefore, to design networks with strong controllability robustness, the critical nodes and edges should be deliberately hidden. This finding is consistent with, and also extends the applicability of, the previous findings: 1) dense and

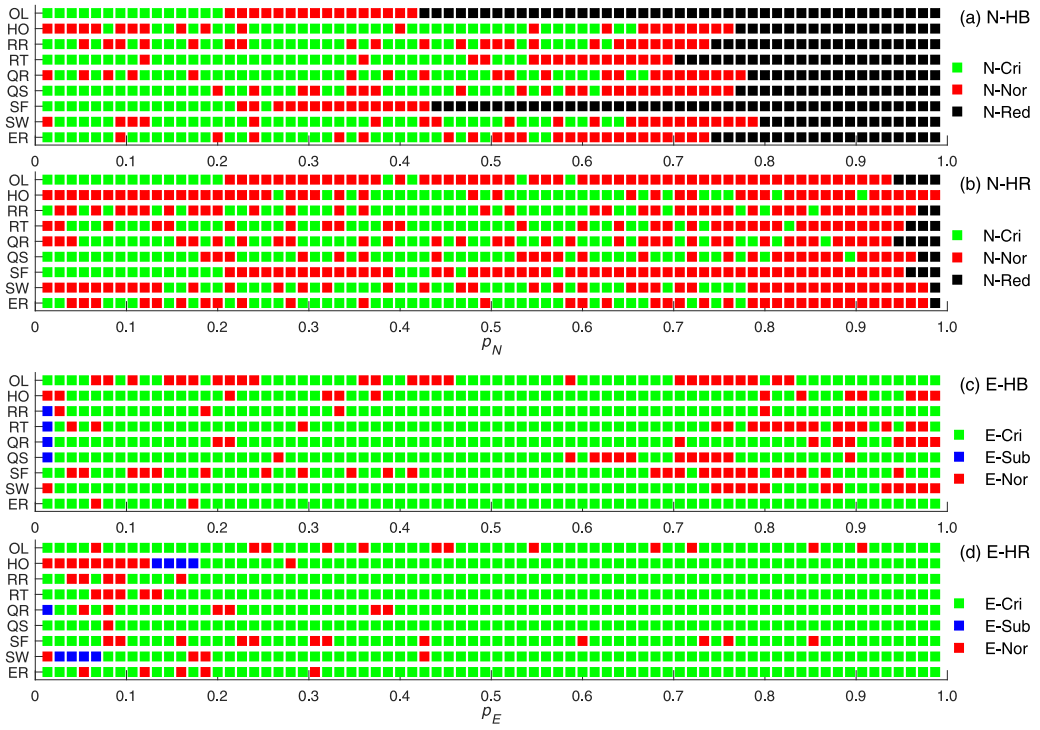


Fig. 10. [color online] Types of the removed nodes under (a) N-HB and (b) N-HR; types of the removed edges under (c) E-HB and (d) E-HR. The initial 'N-' and 'E-' represent node and edge attacks respectively. 'Cri' means 'critical'; 'Nor' means 'normal'; 'Red' means 'redundant'; and 'Sub' means 'subcritical'. The network configuration is $N = 500$ and $\langle k \rangle = 5$.

homogeneous networks have better controllability [5]; 2) extremely-homogeneous topology has the optimal controllability robustness with the fixed numbers of nodes and edges [62].

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this pape

CRediT authorship contribution statement

Yang Lou: Methodology, Validation, Formal analysis, Investigation, Writing - original draft. **Lin Wang:** Methodology, Investigation, Writing - review & editing. **Guanrong Chen:** Methodology, Investigation, Writing - review & editing, Supervision.

Acknowledgments

This research was supported in part by the National Natural Science Foundation of China (No.62002249, 61873167) and in part by the Hong Kong Research Grants Council under the GRF Grant CityU11206320.

References

[1] Newman ME. Networks: An Introduction. Oxford University Press; 2010.
[2] Chen G, Wang X, Li X. Fundamentals of Complex Networks: Models, Structures and Dynamics, 2nd ed. John Wiley & Sons; 2014.
[3] Barabási AL. Network Science. Cambridge University Press; 2016.
[4] Chen G, Lou Y. Naming Game: Models, Simulations and Analysis. Springer; 2019.
[5] Liu Y-Y, Slotine J-J, Barabási AL. Controllability of complex networks. Nature 2011;473(7346):167–73.
[6] Yuan ZZ, Zhao C, Di ZR, Wang W-X, Lai YC. Exact controllability of complex networks. Nature Communications 2013;4:2447.
[7] Pósfai M, Liu Y-Y, Slotine J-J, Barabási AL. Effect of correlations on network controllability. Scientific Reports 2013;3:1067.
[8] Menichetti G, Dall' L, Bianconi G. Network controllability is determined by the density of low in-degree and out-degree nodes. Physical Review Letters 2014;113(7):078701.
[9] Motter AE. Networkcontrology. Chaos: An Interdisciplinary Journal of Nonlinear Science 2015;25(9):097621.
[10] Wang L, Wang X, Chen G, Tang WKS. Controllability of networked MIMO systems. Automatica 2016;69:405–9.
[11] Liu Y-Y, Barabási AL. Control principles of complex systems. Review of Modern Physics 2016;88(3):035006.
[12] Wang L, Wang X, Chen G. Controllability of networked higher-dimensional systems with one-dimensional communication channels. Royal Society Philosophical Transactions A 2017;2088(375):20160215.

[13] Wang L-Z, Chen Y-Z, Wang W-X, Lai YC. Physical controllability of complex networks. Scientific Reports 2017;7:40198.
[14] Zhang Y, Zhou T. Controllability analysis for a networked dynamic system with autonomous subsystems. IEEE Transactions on Automatic Control 2016;62(7):3408–15.
[15] Xiang L, Chen F, Ren W, Chen G. Advances in network controllability. IEEE Circuits and Systems Magazine 2019;19(2):8–32.
[16] Zong G, Yang D. H∞ synchronization of switched complex networks: a switching impulsive control method. Communications in Nonlinear Science and Numerical Simulation 2019;77:338–48.
[17] Holme P, Kim BJ, Yoon CN, Han SK. Attack vulnerability of complex networks. Physical Review E 2002;65(5):056109.
[18] Shargel B, Sayama H, Epstein IR, Bar-Yam Y. Optimization of robustness and connectivity in complex networks. Physical Review Letters 2003;90(6):068701.
[19] Schneider CM, Moreira AA, Andrade JS, Havlin S, Herrmann HJ. Mitigation of malicious attacks on networks. Proceedings of the National Academy of Sciences 2011;108(10):3838–41.
[20] Liu Y-Y, Slotine J-J, Barabási AL. Control centrality and hierarchical structure in complex networks. PLOS ONE 2012;7(9):e44459.
[21] Bashan A, Berezin Y, Buldyrev S, Havlin S. The extreme vulnerability of interdependent spatially embedded networks. Nature Physics 2013;9:667–72.
[22] Xiao Y-D, Lao S-Y, Hou L-L, Bai L. Optimization of robustness of network controllability against malicious attacks. Chinese Physics B 2014;23(11):118902.
[23] Liu J, Zhou M, Wang S, Liu P. A comparative study of network robustness measures. Frontiers of Computer Science 2017;11(4):568–84.
[24] Yamashita K, Yasuda Y, Nakamura R, Ohsaki H. On the predictability of network robustness from spectral measures. 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC) 2019;2:24–9. IEEE
[25] Chen G, Lou Y, Wang L. A comparative study on controllability robustness of complex networks. IEEE Transactions on Circuits and Systems II: Express Briefs 2019;66(5):828–32.
[26] Li X, Zhang Z, Liu J, Gai K. A new complex network robustness attack algorithm. ACM International Symposium on Blockchain and Secure Critical Infrastructure 2019:13–17.
[27] Lou Y, He Y, Wang L, Chen G. Predicting network controllability robustness: A convolutional neural network approach. IEEE Transactions on Cybernetics 2020. doi:10.1109/TCYB.2020.3013251.
[28] Fan C, Zeng L, Sun Y, Liu YY. Finding key players in complex networks through deep reinforcement learning. Nature Machine Intelligence 2020;2:317–24.
[29] Nie T, Guo Z, Zhao K, Lu ZM. New attack strategies for complex networks. Physica A: Statistical Mechanics and its Applications 2015;424:248–53.
[30] Ruan Y-R, Lao S-Y, Wang J-D, Bai L, Chen LD. Node importance measurement based on neighborhood similarity in complex network. Acta Physica Sinica 2017;66(3):038902.
[31] Šimon M, Luptáková ID, Huraj L, Hosťoveckỳ M, Pospíchal J. Combined heuristic attack strategy on complex networks. Mathematical Problems in Engineering 2017;2017.
[32] Yang H, An S. Critical nodes identification in complex networks. Symmetry 2020;12(1):123.
[33] Nguyen Q, Pham H, Cassi D, Bellingeri M. Conditional attack strategy for real-world complex networks. Physica A: Statistical Mechanics and its Applications 2019;530:121561.
[34] Cunha BRd, Gonzalez-Avella JC, Goncalves S. Fast fragmentation of networks using module-based attacks. PLOS ONE 2015;10(11).
[35] Shai S, Kenett DY, Kenett YN, Faust M, Dobson S, Havlin S. Critical tipping point distinguishing two types of transitions in modular network structures. Physical Review E 2015;92(6):062805.
[36] Wang H, Huang J, Xu X, Xiao Y. Damage attack on complex networks. Physica A: Statistical Mechanics and its Applications 2014;408:134–48.
[37] Ma L, Liu J, Duan B. Evolution of network robustness under continuous topological changes. Physica A: Statistical Mechanics and its Applications 2016;451:623–31.
[38] Pu C-L, Pei W-J, Michaelson A. Robustness analysis of network controllability. Physica A: Statistical Mechanics and its Applications 2012;391(18):4420–5.
[39] Nie S, Wang X, Zhang H, Li Q, Wang B. Robustness of controllability for networks based on edge-attack. PLOS ONE 2014;9(2):e89066.
[40] Lu Z-M, Li XF. Attack vulnerability of network controllability. PLOS ONE 2016;11(9).
[41] Sun S, Ma Y, Wu Y, Wang L, Xia C. Towards structural controllability of local-world networks. Physics Letters A 2016;380(22-23):1912–17.
[42] Li X, Chen G. A local-world evolving network model. Physica A: Statistical Mechanics and its Applications 2003;328(1-2):274–86.
[43] Sun P, Kooij RE, He Z, Mieghem PV. Quantifying the robustness of network controllability. International Conference on System Reliability and Safety (ICSRS) 2019:66–76. IEEE
[44] Chen CT. Linear System Theory and Design, 3rd ed. Oxford University Press; 1998.
[45] Lovász L, Plummer MD. Matching theory. American Mathematical Soc 2009;367.
[46] Ruths J, Ruths D. Robustness of network controllability under edge removal. In: Complex Networks IV. Springer; 2013. p. 185–93.
[47] Boldi P, Vigna S. Axioms for centrality. Internet Mathematics 2014;10(3-4):222–62.
[48] Gao Y-L, Chen S-M, Nie S, Ma F, Guan JJ. Robustness analysis of interdependent networks under multiple-attacking strategies. Physica A: Statistical Mechanics and its Applications 2018;496:495–504.
[49] Hao Y, Jia L, Wang Y. Edge attack strategies in interdependent scale-free networks. Physica A: Statistical Mechanics and its Applications 2020;540:122759.
[50] Huang X, Gao J, Buldyrev SV, Havlin S, Stanley HE. Robustness of interdependent networks under targeted attack. Physical Review E 2011;83(6):065101.
[51] Dong G, Gao J, Tian L, Du R, He Y. Percolation of partially interdependent networks under targeted attack. Physical Review E 2012;85(1):016112.
[52] Cui P, Zhu P, Wang K, Xun P, Xia Z. Enhancing robustness of interdependent network by adding connectivity and dependence links. Physica A: Statistical Mechanics and its Applications 2018;497:185–97.
[53] Dong G, Gao J, Du R, Tian L, Stanley HE, Havlin S. Robustness of network of networks under targeted attack. Physical Review E 2013;87(5):052804.
[54] Liu X, Peng H, Gao J. Vulnerability and controllability of networks of networks. Chaos, Solitons & Fractals 2015;80:125–38.
[55] Bellingeri M, Cassi D. Robustness of weighted networks. Physica A: Statistical Mechanics and its Applications 2018;489:47–55.
[56] Erdös P, Rényi A. On the strength of connectedness of a random graph. Acta Mathematica Hungarica 1964;12(1-2):261–7.
[57] Newman ME, Watts DJ. Renormalization group analysis of the small-world network model. Physics Letters A 1999;263(4-6):341–6.
[58] Goh K-I, Kahng B, Kim D. Universal behavior of load distribution in scale-free networks. Physical Review Letters 2001;87(27):278701.
[59] Sorrentino F. Effects of the network structural properties on its controllability. Chaos: An Interdisciplinary Journal of Nonlinear Science 2007;17(3):033101.
[60] Lou Y, Wang L, Chen G. Toward stronger robustness of network controllability: A snapback network model. IEEE Transactions on Circuits and Systems I: Regular Papers 2018;65(9):2983–91.
[61] Lou Y, Wang L, Chen G. Enhancing controllability robustness of q-snapback networks through redirecting edges. Research 2019;2019(7857534).
[62] Lou Y, Wang L, Tsang K-F, Chen G. Towards optimal robustness of network controllability: An empirical necessary condition. IEEE Transactions on Circuits and Systems I: Regular Papers 2020;67(9):3163–74. doi:10.1109/TCSI.2020.2986215.
[63] Herrmann HJ, Schneider CM, Moreira AA, Andrade Jr JS, Havlin S. Onion-like network topology enhances robustness against malicious attacks. Journal of Statistical Mechanics: Theory and Experiment 2011;2011(01):P01027.