Complex Intell. Syst. (2015) 1:37–56
DOI 10.1007/s40747-016-0006-8

ORIGINAL ARTICLE

# Critical nodes identification in complex systems

**Xin Chen**[1]

Received: 18 August 2014 / Accepted: 12 January 2016 / Published online: 6 February 2016


**Abstract** To control complex systems with limited resources, critical nodes need to be identified for protection or removal. Loss of critical nodes decreases or minimizes a system's ability to diffuse entities such as information, goods, or diseases. We design three metrics to assess system homogeneity, diffusion speed, and diffusion scale, and investigate their performance over complex systems. Six algorithms using the three metrics to identify critical nodes are examined. The three nonpolynomial-time algorithms identify the most critical nodes (global optimum). The three polynomial-time algorithms identify critical nodes step by step (local optima), but do not guarantee the global optimum. The three polynomial-time algorithms are compared to other critical nodes identification algorithms and have better performance; they may be applied to practical problems to efficiently identify critical nodes in complex systems.

**Keywords** Complex system · Critical node · Diffusion · Graph theory · Identification for control

## Introduction

Vulnerabilities to natural disasters and terrorist violence require a better understanding of complex systems and new methods of protection and prevention [4]. The objective of this research is to apply informative performance metrics to identify critical nodes in complex systems such as smart grid, social networks, information systems, and criminal organizations. For example, to protect an electrical power grid, most vulnerable transformers and stations whose failures will cause large-scale blackouts (hence they are also critical nodes) need to be provided with backup capacity or enhanced security. For another example, to ensure public safety and security, key members of a criminal or terrorist organization need to be neutralized (e.g., captured or isolated).

Ideally, we would like to protect all nodes in an electrical power grid or an information system. In reality, resources are limited. To protect all nodes in a large system is not affordable. Time to respond to a criminal activity or terrorist attack is short. It is necessary to apply limited resources to the most critical nodes to maximize the effect of either protecting a system or destroying a criminal or terrorist organization. Previous research on critical nodes identification (CNI) suggested a few metrics, but did not indicate why and how they might be useful in practical problems. Some metrics and related CNI algorithms (e.g., [23]) developed earlier can only be applied to systems with special structures. In addition, previous research predominantly studied different CNI algorithms in order to maximize or minimize performance metrics. The characteristics of performance metrics were not studied or used to identify critical nodes.

This research investigates a set of informative performance metrics and applies them to identify critical nodes in complex systems. The main contributions of this research include: (a) development of three new system performance metrics that describe desired properties of complex systems; (b) design of algorithms that use the metrics to identify critical nodes; and (c) analysis of characteristics of the metrics and algorithm complexity.

✉ Xin Chen
xchen@siue.edu

[1] Department of Mechanical and Industrial Engineering, Southern Illinois University, Edwardsville, Illinois 62026-1805, USA

## Background

Nodes in a complex system represent machines, equipment, workstations, computers, generators, control units, operators, and other components each of which is modeled as a separate entity. Edges (links) between nodes represent the flow of entities including products, services, and information. Nodes are linked directly or indirectly. If a node $j$ is linked to a node $j'$ directly, there is an edge between the two nodes. When two nodes $j$ and $j'$ are linked indirectly, there is at least one path between $j$ and $j'$ through other nodes so that entities can be diffused from $j$ to $j'$ and/or from $j'$ to $j$. When two nodes are not linked, there is no path between the two nodes. Entities may flow along both directions between two nodes connected by an undirected edge [10,11,13]. Directed edges are also called arcs; entities may flow along only one direction between two nodes connected by an arc.

There have been many studies on CNI, and complex networks and complex systems in general [29]. Most recently, Shen and Smith [23] used dynamic programming algorithms to identify critical nodes in trees and series–parallel systems. The performance metrics of interest were: (a) the number of components, which was to be maximized; and (b) the largest component order (i.e., the number of nodes in a component), which was to be minimized. For systems which can be modeled as trees and series-parallel graphs, the complexity of the dynamic programming algorithms was at most $O(n^5 \log n)$. These algorithms may be extended to generalized connected systems, which are interpreted as $k$-hole systems. The complexity of the algorithms, however, increases exponentially as $k$ increases. The algorithms are not applicable to systems with disconnected components.

Analysis of system vulnerability is related to CNI. Dinh et al. [16] used dynamic programming to identify critical nodes and links. The algorithms developed were approximation algorithms and the complexity was at most $O(\log^{1.5} n)$. The performance metric of interest was pairwise connectivity. The pairwise connectivity between two nodes is one if this pair is connected and zero otherwise. In an undirected system, a pair of nodes is connected if and only if there exists at least one path (in both directions) between the two nodes. The system pairwise connectivity is the sum of pairwise connectivity between any pair of nodes. For a given level of degradation in pairwise connectivity, it was shown that to find a minimum set of nodes or edges (called $\beta$-disruptor), whose removal causes the specific level of degradation, was an NP-complete problem for undirected systems.

CNI was also studied in the context of system reliability [28]. A reliability metric, $\Pi$, was used to describe the average reliability between every pair of nodes in a system. An evolutionary algorithm was used to identify critical components (nodes), the removal of which aimed to minimize both $\Pi$ and the cost of incapacitating links (multiple-objective optimization). The evolutionary algorithm provided good-quality solutions; its Pareto fronts (efficient frontiers) were close to the real Pareto optimal solutions.

In the study of the Internet and World Wide Web [1,7] and more generally complex systems such as airline routes, electric power grids, and disease propagation [8], the diameter of a system, which is the average length of the shortest paths between any two nodes in the system, was analyzed when nodes were removed according to the number of links they have; nodes with more links were removed first. Compared to removing randomly selected nodes, the link-based node removal increased the diameter faster, although the diameter does not always increase when a node is removed.

Earlier studies on CNI focused on general systems and algorithm complexity. For a fixed system property (i.e., a value or a range of values for a performance metric), a node-deletion problem aims to find the minimum number of nodes which must be deleted from a given system so that the result satisfies the property. It was shown that the node-deletion problem for a system property is NP-hard or NP-complete if the property is nontrivial and hereditary [19,25]. A system property is nontrivial if it is true for infinitely many systems and false for infinitely many systems. A property is hereditary if for any system satisfying the property, all nodes-induced subsystems also satisfy the property.

Results of recent studies on CNI validate the node-deletion problem analyzed more than 30 years ago. The largest component order in a system [23] is nontrivial and hereditary. Testing (certificate-checking) for the largest component order can be performed in polynomial time. For instance, a depth-first algorithm such as Tarjan's algorithm may be used to identify the largest component order with the complexity of $O(\text{nodes} + \text{edges})$. A system with $n$ nodes has at most $\frac{n(n-1)}{2}$ edges. The certificate-checking is therefore bounded by $O(n^2)$. When the largest component order is the performance metric of interest, to identify the most critical nodes, whose removal minimizes the largest component size, is NP-complete according to Lewis and Yannakakis [19].

Similarly, the system pairwise connectivity [16] is nontrivial and hereditary. Tarjan's algorithm may be used to find pairwise connectivity between any pair of nodes and further calculate the pairwise connectivity. The CNI problem is therefore NP-complete when the system pairwise connectivity is of concern. Not all system performance metrics are hereditary, however. The system diameter [1] is not hereditary. Deleting a node may decrease the diameter although it increases the diameter most of the time. The CNI problem might belong to class P if the objective is to maximize the system diameter.

Some other recent work is related to CNI. In physics and engineering, research was focused on developing mathematical models and analyzing system properties [2,12,22]. In social science, many measures including centrality [17], complement [14], and reciprocal [5] were developed to describe system properties. In industrial process monitoring and control where multiple processes form a complex manufacturing system, data-driven approaches were applied for fault prognostics and diagnostics [26,27]. Borgatti [5] defined two types of problems to assess the importance of nodes. The Key Player Problem-Positive studied the extent to which a node is embedded in the system. The Key Player Problem-Negative studied the amount of reduction in cohesiveness of a system after elimination of a node. The Dynamic Network Modeling (DNA) [4,9,20,24] was developed to model and analyze complex systems. The DNA was successfully applied to terrorist networks and used to identify critical nodes through simulation experiments.

In summary, previous research on CNI validated certain analytical results and alluded to their applications to practical problems, but failed to design metrics that are meaningful for practical problems. There are two types of CNI problems: the optimization problem and the recognition problem. In the optimization problem, given limited resources (i.e., a certain number of nodes need to be protected or removed), which nodes' removal can minimize or maximize a performance metric? In the recognition problem, given a desired property (e.g., $\geq$ the value of a performance metric or $\leq$ the value of a performance metric), what is the minimum number of nodes that need to be protected or removed to satisfy the property? These two types of problems are equivalent in terms of algorithm complexity. This research is focused on the optimization problem, namely, to identify the most critical nodes. What was missing in previous research, which is the focus of this study, lies in two areas:

(a) Which properties of a complex system need to be measured? A set of performance metrics must be designed to assess the impact of removing a portion of nodes from a complex system. Previous research suggested a few metrics, but did not indicate why and how they might be useful in practical problems. In addition, some metrics and related CNI algorithms (e.g., [23]) can only be applied to systems with special structures. This research designs three metrics that indicate a complex system's speed and scale of diffusing entities, which are useful in many real-world application; and
(b) What are the characteristics of performance metrics? Previous research predominantly studied different CNI algorithms in order to maximize or minimize performance metrics. The characteristics of performance metrics affect the efficiency and effectiveness of CNI algorithms. Moreover, performance metrics themselves can be used to identify critical nodes. This research identifies the values for each of the three new performance metrics and their corresponding network structures of a complex system, which provide important insight into how these metrics may be used to identify critical nodes.

## Problem definition

Let $G(V, E)$ represent a system of $n$ vertices (nodes) and $s$ edges. $n$ and $s$ are the order and size of $G(V, E)$, respectively. $V$ is a set of all nodes in $G(V, E) : v_1, v_2, \ldots, v_n$. $E$ is a set of all edges in $G(V, E) : e_1, e_2, \ldots, e_s$. A path from $v_i$ to $v_j$ is a set of nodes and edges that connect $v_i$ and $v_j$, with which entities may be diffused from $v_i$ to $v_j$. In an undirected $G(V, E)$, any edge is bidirectional, or undirected; a path from $v_i$ to $v_j$ is also a path from $v_j$ to $v_i$. A directed $G(V, E)$ has at least one directed edge (arc). An arc from $v_i$ to $v_j$ connects $v_i$ and $v_j$; entities may be diffused from $v_i$ to $v_j$, but not from $v_j$ to $v_i$. Since information is diffused in both directions in most social and information networks, this research focuses on CNI for undirected systems.

$G(V, E)$'s ability to diffuse entities such as information, goods, or diseases [15] is reflected in homogeneity, diffusion speed, and diffusion scale, all of which are performance metrics that describe the properties of $G(V, E)$ [3]. In a system with high homogeneity (i.e., a homogeneous network), relationships between nodes are the same or similar. In a network with low homogeneity (i.e., a heterogeneous network), relationships between nodes are much different. Diffusion speed indicates how fast entities are diffused between nodes. Diffusion scale indicates how many nodes in $G(V, E)$ diffuse (or receive) entities to (or from) other nodes. This research investigates using performance metrics to identify critical nodes in $G(V, E)$; removal of critical nodes minimizes homogeneity, diffusion speed, and/or diffusion scale. When resources are limited, CNI helps control systems in order to, for example,

(a) minimize the possibility and scale of criminal organizations or terrorist attacks by neutralizing the most critical criminals or terrorists;
(b) maximize a computer network's resilience to incidents (e.g., cyber-attacks) and accidents (e.g., random computer failures) by protecting the most critical nodes such as routers and servers; and
(c) minimize disruptions in sensor or logistics networks by providing backup capacity to the most critical nodes.

## Performance metrics

### Homogeneity: normalized expected geodesic distance

Nodes $v_i$ and $v_j$ are neighbors if an edge connects $v_i$ and $v_j$. The geodesic distance, $d_{v_i,v_j}$, is the distance of the shortest path(s) between $v_i$ and $v_j$. If $v_i$ and $v_j$ are neighbors, $d_{v_i,v_j} =$

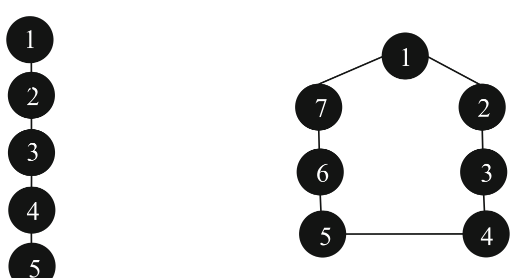
1. If there exists no path between $v_i$ and $v_j$, $d_{v_i,v_j} = 0$. Suppose $v_i$ diffuses entities to other nodes in a system with a total of $n$ nodes, the expected geodesic distance (EGD) from $v_i$ to other nodes is $\frac{\sum_{j=1, j\neq i}^{n} d_{v_i,v_j}}{n-1}$. Further, suppose all nodes, $v_1, v_2, \ldots, v_n$, have equal probability, $\frac{1}{n}$, to be the source node that begins the diffusion of entities to other nodes, the EGD of the network is $\text{EGD} = \frac{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} d_{v_i,v_j}}{\frac{n(n-1)}{2}}$.

At each step of diffusion, a node diffuses entities to all its neighbors. The EGD indicates the expected number of steps it takes to diffuse entities from the source node to other nodes. If diffusion time is proportional to the number of steps, the larger the EGD is, the longer is the total diffusion time and the lower is the diffusion speed. Since $0 \leq d_{v_i,v_j} \leq n-1$, $0 \leq \text{EGD} \leq n-1$. $\text{EGD} = 1$ for a clique, which is a fully connected network (i.e., $d_{v_i,v_j} = 1$ for $\forall v_i, v_j$). $\text{EGD} = 0$ for a fully disconnected network without any edge (i.e., $s = 0$). In a connected network, $d_{v_i,v_j} \geq 1$ for $\forall v_i, v_j$; in a disconnected network, $\exists v_i, v_j$ such that $d_{v_i,v_j} = 0$. Figure 1 depicts two systems 1(a) and 1(b). For 1(a), $\text{EGD} = \frac{(1+2+3+4)+(1+2+3)+(1+2)+1}{\frac{5(5-1)}{2}} = 2$. For 1(b), $\text{EGD} = \frac{(1+2+3+1+2+3)+(1+2+3+2+3)+(1+2+3+3)+(1+2+3)+(1+2)+1}{\frac{7(7-1)}{2}} = 2$. Both systems have the same EGD (i.e., on average, it takes the same amount of time (two steps) to diffuse entities).

The EGD indirectly measures a network's diffusion speed and does not take into consideration the order of the network, $n$. For instance, entities can be diffused to total five nodes in 1(a) ($n = 5$), whereas they can be diffused to total seven nodes in 1(b) ($n = 7$). Moreover, 1(b) is more homogenous than 1(a). The largest geodesic distance (LGD) of 1(b) is three, whereas the LGD of 1(a) is four (the distance of the shortest path between nodes 1 and 5). System 1(a) is less homogenous than 1(b) because there is a larger difference between the smallest geodesic distance and LGD in 1(a) than in 1(b).

To accurately measure a system's homogeneity, the normalized expected geodesic distance (NEGD) is defined in Eq. (1) for $n > 1$. In essence, $\text{NEGD} = \frac{\text{EGD}}{\text{LGD}}$. $\text{NEGD} = 0.500$ for 1(a) and $\text{NEGD} = 0.667$ for 1(b). The larger the NEGD is, the more homogeneous is a system. $\text{NEGD} = 1$ for a clique. For a fully disconnected system, it is defined that $\text{NEGD} = 0$. Although such a system is homogeneous, it is the most desirable case when the goal is to destroy a system and the worst case when the goal is to protect a system. Let $\theta(v_i)$ be the degree of $v_i$, the number of edges that connect $v_i$ and its neighbors. The NEGD of $G(V, E)$ is summarized in Proposition 1.

$$\text{NEGD} = \frac{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} d_{v_i,v_j}}{\frac{n(n-1)\text{LGD}}{2}} \tag{1}$$




**(a)** A system of five nodes. **(b)** A system of seven nodes.

**Fig. 1** Two systems with the same EGD

**Proposition 1** *Suppose $G(V, E)$ has total $n$ nodes. Let $n_f$ ($0 \leq n_f \leq n$) be the number of fully connected nodes $v_i$s. $\theta(v_i) = n-1$. Table 1 depicts the NEGD of $G(V, E)$. $0 \leq \text{NEGD} \leq 1$. $\text{NEGD} \leq c$, $0 \leq c \leq 1$, is a nontrivial property of $G(V, E)$. $\text{NEGD} \leq c$, $c = 0$ or $c = 1$, is a hereditary property of $G(V, E)$. $\text{NEGD} \leq c$, $0 < c < 1$, is a nonhereditary property of $G(V, E)$.*

*Proof of Proposition 1* When $n = 1$, $\text{NEGD} = 0$ per definition.
When $n = 2$, $\text{NEGD} = 0$ if there is no edge; $\text{NEGD} = 1$ if there is one edge.
When $n \geq 3$, let $V_f$ represent a set of nodes, each of which is connected to other $n-1$ nodes in $G(V, E)$. $\theta(v_i) = n-1$ for $\forall v_i \in V_f$. $\theta(v_j) < n-1$ for $\forall v_j \notin V_f$. $V_f$ has $n_f$ nodes; $0 \leq n_f \leq n$.

(a) $n_f = n$. $G(V, E)$ is a clique and $\text{NEGD} = 1$;
(b) $n_f \neq n-1$. If $n_f = n-1$, there is only one node, say $v_j \in V$, such that $v_j \notin V_f$ (i.e., $\theta(v_j) < n-1$). Since all $n-1$ nodes in $V_f$ have the degree of $n-1$, all $n-1$ nodes in $V_f$ must be connected to $v_j$. This indicates that $\theta(v_j) = n-1$. This is contradictory to the condition that $v_j \notin V_f$. Therefore $n_f \neq n-1$;
(c) $1 \leq n_f \leq n-2$. $\text{NEGD} \leq \frac{\frac{n_f(n_f-1)}{2} + n_f(n-n_f) + 2\frac{(n-n_f)(n-n_f-1)}{2}}{2\frac{n(n-1)}{2}} = 1 - \frac{n_f(2n-n_f-1)}{2n(n-1)} = \frac{1}{2n(n-1)}\left[\left(\frac{2n-1}{2} - n_f\right)^2 + \frac{4n^2-4n-1}{4}\right]$. Since $1 \leq n_f \leq n-2 < \frac{2n-1}{2}$, $\text{NEGD} \leq \frac{n-1}{n}$ when $n_f = 1$. Since $\lim_{n\to\infty} \frac{n-1}{n} = 1$, $\text{NEGD} < 1$. $\text{NEGD} > \frac{\frac{n_f(n_f-1)}{2} + n_f(n-n_f) + \frac{(n-n_f)(n-n_f-1)}{2}}{2\frac{n(n-1)}{2}} = \frac{1}{2}$. $\frac{1}{2} < \text{NEGD} < 1$; and
(d) $n_f = 0$. Three different situations can be analyzed: $\text{LGD} = n-1$, $1 \leq \text{LGD} \leq n-2$, and $\text{LGD} = 0$.

   (i) $\text{LGD} = n-1$. If $n = 3$, then $n_f = 1$, which is contradictory to $n_f = 0$. Therefore $n \geq 4$. $G(V, E)$ comprised a chain of $n$ nodes. (An example is shown in Fig. 1a) $\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{v_i,v_j} = [1 + \cdots + (n-1)] + [1 + \cdots + (n-2)] + \cdots +$

**Table 1** NEGD of $G(V, E)$ with total $n$ nodes and $n_f$ fully connected nodes

| $G(V, E)$ | | | NEGD | |
|---|---|---|---|---|
| | | | Value/range | $n \to \infty$ |
| $n = 1$ | | | 0 | |
| $n = 2$ | $n_f = n$ | | 1 | |
| | $n_f = 0$ | | 0 | |
| $n \geq 3, n_f \neq n - 1$ | $n_f = n$ | | 1 | |
| | $1 \leq n_f \leq n - 2$ | | $\left(\frac{1}{2}, 1 - \frac{n_f(2n-n_f-1)}{2n(n-1)}\right]$ | $(\frac{1}{2}, 1)$ |
| | $n_f = 0$ | $\text{LGD} = n - 1, \quad n \neq 3$ | $\frac{1}{3} + \frac{2}{3(n-1)}$ | $\left(\frac{1}{3}, \frac{5}{9}\right]$ |
| | | $1 \leq \text{LGD} \leq n - 2$ | $\left[\frac{(\text{LGD}+1)(\text{LGD}+2)}{3n(n-1)}, 1 - \frac{2(\text{LGD}+1)(\text{LGD}-1)}{3n(n-1)}\right)$ | $(0, 1)$ |
| | | $\text{LGD} = 0$ | 0 | |

$[1+2]+1 = \frac{n(n-1)}{2} + \frac{(n-1)(n-2)}{2} + \cdots + \frac{3\times 2}{2} + \frac{2\times 1}{2} = \frac{1}{2}\left[((n-1)^2 + (n-2)^2 + \cdots + 2^2 + 1^2) + ((n-1) + (n-2) + \cdots + 2 + 1)\right] = \frac{1}{2}\left[\frac{(n-1)n(2n-2+1)}{6} + \frac{n(n-1)}{2}\right] = \frac{(n-1)n(n+1)}{6}$. NEGD $= \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{v_i,v_j}}{\frac{n(n-1)}{2}(n-1)} = \frac{\frac{(n-1)n(n+1)}{6}}{\frac{n(n-1)}{2}(n-1)} = \frac{n+1}{3(n-1)} = \frac{1}{3} + \frac{2}{3(n-1)}$. NEGD is strictly monotonically decreasing. When $n = 4$, NEGD $= \frac{5}{9}$. Since $\lim_{n\to\infty}\left[\frac{1}{3} + \frac{2}{3(n-1)}\right] = \frac{1}{3}$, $\frac{1}{3} <$ NEGD $\leq \frac{5}{9}$;

(ii) $1 \leq \text{LGD} \leq n-2$. There exists at least one chain of nodes, say $V_1$; the distance of the shortest path between the two end nodes of the chain is LGD. $V_1$ has LGD + 1 nodes. NEGD $= \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{v_i,v_j}}{\frac{n(n-1)}{2}\text{LGD}} = \frac{\frac{\text{LGD}(\text{LGD}+1)(\text{LGD}+2)}{6} + \sum_{i,j} d_{v_i,v_j}}{\frac{n(n-1)}{2}\text{LGD}} \geq \frac{\frac{\text{LGD}(\text{LGD}+1)(\text{LGD}+2)}{6}}{\frac{n(n-1)}{2}\text{LGD}} = \frac{(\text{LGD}+1)(\text{LGD}+2)}{3n(n-1)} \geq \frac{2}{n(n-1)}$. Since $\lim_{n\to\infty} \frac{2}{n(n-1)} = 0$, NEGD $> 0$. Meantime, NEGD $= \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} d_{v_i,v_j}}{\frac{n(n-1)}{2}\text{LGD}}$

$= \frac{\frac{\text{LGD}(\text{LGD}+1)(\text{LGD}+2)}{6} + \sum_{i,j} d_{v_i,v_j}}{\frac{n(n-1)}{2}\text{LGD}}$

$< \frac{\frac{\text{LGD}(\text{LGD}+1)(\text{LGD}+2)}{6} + \left[\frac{n(n-1)}{2} - \frac{\text{LGD}(\text{LGD}+1)}{2}\right]\text{LGD}}{\frac{n(n-1)}{2}\text{LGD}} = 1 - \frac{2(\text{LGD}+1)(\text{LGD}-1)}{3n(n-1)} \leq 1$; and

(iii) $\text{LGD} = 0$. NEGD $= 0$ per definition.

The desired system property in the recognition problem is NEGD $\leq c$, where $0 \leq c \leq 1$. This property is nontrivial since it is true for infinitely many systems and false for infinitely many systems for given $c$. NEGD $\leq 1$ is hereditary since the maximum of NEGD is one. NEGD $\leq 0$ if and only if $G(V, E)$ does not have any edge; any subsystem of $G(V, E)$ does not have any edge. NEGD $\leq 0$ is also hereditary. When $0 < c < 1$, NEGD $\leq c$ is nonhereditary. The proof is as follows.

For any value of $c$, $0 < c < 1$, $G(V, E)$ may be constructed such that it includes only one edge and NEGD $= \frac{2}{n(n-1)} \leq c < 1, n \geq 3$. A subsystem of $G(V, E)$, which has two nodes and one edge that connects the two nodes, may be produced by removing all nodes without any edge. For this subsystem, NEGD $= 1 > c$. NEGD $\leq c$, $0 < c < 1$, is therefore nonhereditary.

This concludes the proof of Proposition 1. □

Proposition 1 reveals the homogeneity of $G(V, E)$ ($n \geq 3$):

(a) If $G(V, E)$ is a clique (i.e., $n_f = n$), NEGD $= 1$. A clique is the most homogeneous among all system structures. NEGD remains one regardless of which nodes are removed from $G(V, E)$ (hereditary). All nodes in $G(V, E)$ are equally critical;
(b) If $G(V, E)$ has at least one node that is fully connected but is not a clique (i.e., $1 \leq n_f \leq n - 2$), NEGD is greater than 0.5 and less than one;
(c) If $G(V, E)$ does not have any node that is fully connected (i.e., $n_f = 0$), but comprised a chain of $n$ nodes (i.e., LGD $= n - 1$), NEGD is described by a closed form, NEGD $= \frac{1}{3} + \frac{2}{3(n-1)}$. NEGD is strictly monotonically decreasing as $n$ increases. As a chain becomes longer, it is less homogeneous;
(d) If $G(V, E)$ does not have any node that is fully connected (i.e., $n_f = 0$), and does not comprise a chain of $n$ nodes (i.e., $1 \leq \text{LGD} \leq n - 2$), NEGD is greater than zero and less than one. Most real-world networks follow this structure ($n_f = 0$ and $1 \leq \text{LGD} \leq n-2$) and NEGD can be used to describe their homogeneity and help identify critical nodes;
(e) If $G(V, E)$ does not have any edge (i.e., LGD $= 0$), NEGD $= 0$; and
(f) Compared to the NEGD ($\frac{1}{3} <$ NEGD $\leq \frac{1}{2}$) of $G(V, E)$ ($n \geq 5$) that comprised a chain, the NEGD ($\frac{1}{2} <$

**Table 2** NEMS of $G(V, E)$ with total $n$ nodes and $n_f$ fully connected nodes

| $G(V, E)$ | | | NEMS | |
|---|---|---|---|---|
| | | | Value/range | $n \to \infty$ |
| $n = 1$ | | | 0 | |
| $n = 2$ | $n_f = n$ | | 1 | |
| | $n_f = 0$ | | 0 | |
| $n \geq 3, \quad n_f \neq n - 1$ | $n_f = n$ | | 1 | |
| | $1 \leq n_f \leq n - 2$ | | 0.5 | |
| | $n_f = 0$ | $\mathrm{LGD} = n - 1, \quad n \neq 3$ | $\frac{1}{n-1}$ | $\left(0, \frac{1}{3}\right]$ |
| | | $1 \leq \mathrm{LGD} \leq n - 2$ | $\left[\frac{\mathrm{LGD}+1}{n(n-1)}, \frac{n-2}{n}\right]$ | $(0, 1)$ |
| | | $\mathrm{LGD} = 0$ | 0 | |

NEGD $\leq$ 1) of $G(V, E)$ that has at least one fully connected node is higher. Systems with at least one fully connected node are more homogeneous than those with the chain structure.

## Diffusion speed: normalized expected minimum speed

$G(V, E)$ comprises components. A component is disconnected from all other components in $G(V, E)$; there are no edges that connect nodes in one component and nodes in another component. Nodes in the same component are connected to each other directly or indirectly. $G(V, E)$ has only one component if it is a clique. A fully disconnected $G(V, E)$ of $n$ nodes comprised $n$ components, each of which has one node. At each step of diffusion, a node $v_i$ diffuses entities to all its neighbors. After one or more steps, entities are diffused to all nodes in the component to which $v_i$ belongs. Let $\mathrm{LGD}_{v_i}$ be the largest geodesic distance of the component to which $v_i$ belongs. $\mathrm{LGD}_{v_i}$ is the maximum number of steps required for all nodes in the component to receive entities diffused from a randomly selected node $v_i$. Let $n_{v_i}$ be the total number of nodes in the component to which $v_i$ belongs; $n_{v_i}$ is the order of the component. $n_{v_i} - 1$ is the total number of nodes that receive entities diffused from $v_i$. $\frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}$ is the expected minimum number of nodes that receive entities diffused from $v_i$ at each step. $\frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}$ is the minimum diffusion speed of the component to which $v_i$ belongs. All nodes in the same component have the same minimum diffusion speed. $\frac{\sum_{i=1}^{n} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}}{n}$ is the expected minimum speed, which indicates the expected minimum number of nodes in $G(V, E)$ that receive entities at each step of diffusion. The maximum value of $\frac{\sum_{i=1}^{n} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}}{n}$ is $n-1$, which is for a clique. To compare systems of different orders, the normalized expected minimum speed (NEMS) is defined in Eq. (2) for $n > 1$. NEMS $= 0$ for a fully disconnected $G(V, E)$ and NEMS $= 1$ for a clique. The larger the NEMS is, the higher diffusion speed does $G(V, E)$ have. For the two systems in Fig. 1, NEMS $= 0.250$ for 1(a) and NEMS $= 0.333$ for 1(b); system 1(b) has higher speed than 1(a).

$$\mathrm{NEMS} = \frac{\sum_{i=1}^{n} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}}{n(n-1)} \tag{2}$$

**Proposition 2** *Suppose $G(V, E)$ has total $n$ nodes and $n_f$ ($0 \leq n_f \leq n$) fully connected nodes. Table 2 depicts the* NEMS *of $G(V, E)$.* $0 \leq$ NEMS $\leq 1$. NEMS $\leq c$, *$0 \leq c \leq 1$, is a nontrivial property of $G(V, E)$.* NEMS $\leq c$, *$c = 0$ or $c = 1$, is a hereditary property of $G(V, E)$.* NEMS $\leq c$, *$0 < c < 1$, is a nonhereditary property of $G(V, E)$.*

*Proof of Proposition 2* When $n = 1$, NEMS $= 0$ per definition.
When $n = 2$, NEMS $= 0$ if there is no edge; NEMS $= 1$ if there is one edge.
When $n \geq 3$,

(a) $n_f = n$. $G(V, E)$ is a clique and NEMS $= 1$;
(b) $n_f \neq n - 1$ (see proof in Proposition 1);
(c) $1 \leq n_f \leq n - 2$. NEMS $= \frac{\sum_{i=1}^{n} \frac{n-1}{2}}{n(n-1)} = \frac{1}{2}$; and
(d) $n_f = 0$. Three different situations can be analyzed: LGD $= n - 1$, $1 \leq$ LGD $\leq n - 2$, and LGD $= 0$.
   (i) LGD $= n - 1$. $n \geq 4$ (see proof in Proposition 1). $G(V, E)$ comprised a chain of $n$ nodes. NEMS $= \frac{\sum_{i=1}^{n} \frac{n-1}{n-1}}{n(n-1)} = \frac{1}{n-1}$. NEMS is strictly monotonically decreasing. When $n = 4$, NEMS $= \frac{1}{3}$. Since $\lim_{n\to\infty} \frac{1}{n-1} = 0$, $0 <$ NEMS $\leq \frac{1}{3}$;
   (ii) $1 \leq$ LGD $\leq n - 2$. There exists at least one chain of nodes; the distance of the shortest path between the two end nodes of the chain is LGD. Suppose this chain of nodes belongs to a component $V_1$, which has $n_1$ nodes. $n_1 \geq$ LGD $+ 1$. NEMS $= \frac{\sum_{i=1}^{n} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}}{n(n-1)} = \frac{\sum_{V_1} \frac{n_1-1}{\mathrm{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}}{n(n-1)}$. The minimum value of $\sum_{V-V_1} \frac{n_{v_i}-1}{\mathrm{LGD}_{v_i}}$ is zero; each node $v_i$, $v_i \notin$

$V_1$, is disconnected from other nodes. $\sum_{V_1} \frac{n_1-1}{\text{LGD}} = \frac{n_1(n_1-1)}{\text{LGD}} \geq \text{LGD}+1$. Therefore, $\text{NEMS} \geq \frac{\text{LGD}+1}{n(n-1)}$. When $\text{NEMS} = \frac{\text{LGD}+1}{n(n-1)}$, $G(V, E)$ comprised $n - \text{LGD}$ components. One component has $\text{LGD} + 1$ nodes, which form a chain; the other $n - (\text{LGD} + 1)$ components each have one node. When $\text{LGD} = 1$, $\lim_{n\to\infty} \frac{\text{LGD}+1}{n(n-1)} = \lim_{n\to\infty} \frac{2}{n(n-1)} = 0$. When $\text{LGD} = n-2$, $\lim_{n\to\infty} \frac{\text{LGD}+1}{n(n-1)} = \lim_{n\to\infty} \frac{n-1}{n(n-1)} = \lim_{n\to\infty} \frac{1}{n} = 0$. $\text{NEMS} > 0$.

To find the maximum of NEMS, note that $\sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq \sum_{V-V_1}(n_{v_i} - 1)$. When $\sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} = \sum_{V-V_1}(n_{v_i} - 1)$, each node $v_i$, $v_i \notin V_1$, belongs to a clique and $\text{LGD}_{v_i} = 1$. Since there are total $n - n_1$ nodes $v_i's$ such that $v_i \notin V_1$, $\sum_{V-V_1}(n_{v_i} - 1) = \sum_{V-V_1} n_{v_i} - (n - n_1) \leq (n - n_1)^2 - (n - n_1)$. $\sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq (n - n_1)^2 - (n - n_1)$. When $\sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} = (n - n_1)^2 - (n - n_1)$, not only each node $v_i$, $v_i \notin V_1$, belongs to a clique, but all $v_i's$, $v_i \notin V_1$, belong to the same clique. $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq \frac{n_1(n_1-1)}{\text{LGD}} + (n - n_1)^2 - (n - n_1) = \frac{n_1[(1+\text{LGD})n_1-(2\text{LGD}\cdot n+1-\text{LGD})]+\text{LGD}(n-1)n}{\text{LGD}}$. If $n_1 = n$, $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq \frac{n(n-1)}{\text{LGD}}$; $G(V, E)$ has one component. If $\text{LGD} = 1$, $G(V, E)$ becomes a clique, which is contradictory to the condition that $n_f = 0$. $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq \frac{n(n-1)}{\text{LGD}} \leq \frac{n(n-1)}{2}$. If $n = 3$ and $\text{LGD} = 2$, $\text{LGD} = n-1$, which is contradictory to the condition that $\text{LGD} \leq n - 2$. $n \geq 4$ when $\text{LGD} = 2$. $G(V, E)$ has a single component of $n$ ($n \geq 4$) nodes with $\text{LGD} = 2$ when $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} = \frac{n(n-1)}{2}$.

If $n_1 < n$, $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq n_1(n_1 - 2n + 1) + n(n-1) + \frac{n_1(n_1-1)}{\text{LGD}}$. When $\text{LGD} = 1$, $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq n(n-1) - 2n_1(n - n_1)$; $G(V, E)$ comprised two cliques. To further maximize $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}}$, note that $n(n-1) - 2n_1(n - n_1) = n(n-1) + 2(n_1 - \frac{n}{2})^2 - \frac{n^2}{2}$. When $n_1 = n-1$, $n(n-1) + 2(n_1 - \frac{n}{2})^2 - \frac{n^2}{2} \leq (n-1)(n-2)$. When $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} = (n-1)(n-2)$, $G(V, E)$ comprised two components: one is a clique with $n - 1$ nodes and the other has one node. $\frac{n(n-1)}{2} = (n-1)(n-2)$ when $n = 4$; $\frac{n(n-1)}{2} < (n-1)(n-2)$ when $n > 4$. Therefore, $\sum_{V_1} \frac{n_1-1}{\text{LGD}} + \sum_{V-V_1} \frac{n_{v_i}-1}{\text{LGD}_{v_i}} \leq (n-1)(n-2)$. $\text{NEMS} \leq \frac{(n-1)(n-2)}{n(n-1)} = \frac{n-2}{n}$. $\lim_{n\to\infty} \frac{n-2}{n} = 1$. $\text{NEMS} < 1$.

In summary, $\frac{\text{LGD}+1}{n(n-1)} \leq \text{NEMS} \leq \frac{n-2}{n}$. When $\text{NEMS} = \frac{\text{LGD}+1}{n(n-1)}$, $G(V, E)$ comprised a chain of $\text{LGD} + 1$ nodes, and $n - (\text{LGD} + 1)$ components each have one node. When $\text{NEMS} = \frac{n-2}{n}$, $G(V, E)$ comprised a clique of $n - 1$ nodes and another component of one node; and

(iii) $\text{LGD} = 0$. $\text{NEMS} = 0$ per definition.

The desired system property in the recognition problem is $\text{NEMS} \leq c$, where $0 \leq c \leq 1$. This property is nontrivial since it is true for infinitely many systems and false for infinitely many systems for given $c$. $\text{NEMS} \leq 1$ is hereditary since the maximum of NEMS is one. $\text{NEMS} \leq 0$ if and only if $G(V, E)$ does not have any edge; any subsystem of $G(V, E)$ does not have any edge. $\text{NEMS} \leq 0$ is also hereditary. When $0 < c < 1$, $\text{NEMS} \leq c$ is nonhereditary. The proof is as follows:

For any value of $c$, $0 < c < 1$, $G(V, E)$ can be constructed such that it is a chain system and $\text{NEGD} = \frac{1}{n-1} \leq c < 1$, $n \geq 4$. A subsystem of $G(V, E)$, which has two nodes and one edge that connects the two nodes, may be produced by removing nodes on either end of the chain until only two nodes are left. For this subsystem, $\text{NEMS} = 1 > c$. $\text{NEMS} \leq c$, $0 < c < 1$, is therefore nonhereditary.

This concludes the proof of Proposition 2. □

Proposition 2 reveals the diffusion speed of $G(V, E)$ ($n \geq 3$):

(a) If $G(V, E)$ is a clique (i.e., $n_f = n$), $\text{NEMS} = 1$. A clique has the highest diffusion speed among all system structures. Regardless of which source node that diffuses entities, all other nodes in $G(V, E)$ receive the entities in one step. NEMS remains one regardless of which nodes are removed from $G(V, E)$. All nodes in $G(V, E)$ are equally critical;

(b) If $G(V, E)$ comprised a single component, $\text{NEMS} = \frac{1}{\text{LGD}}$. When $G(V, E)$ has at least one node that is fully connected (i.e., $1 \leq n_f \leq n - 2$), $G(V, E)$ comprised a single component and $\text{LGD} = 2$; $\text{NEMS} = 0.5$. When $G(V, E)$ does not have any node that is fully connected (i.e., $n_f = 0$), but comprised a chain of $n$ nodes (i.e., $\text{LGD} = n - 1$), $G(V, E)$ comprised a single component and $\text{NEMS} = \frac{1}{n-1}$. In this case, NEMS is strictly monotonically decreasing as $n$ increases. As a chain becomes longer, its diffusion speed decreases;

(c) If $G(V, E)$ does not have any node that is fully connected (i.e., $n_f = 0$), and does not comprise a chain of $n$ nodes (i.e., $1 \leq \text{LGD} \leq n-2$), NEMS is greater than zero and less than one. NEMS approaches zero as $n$ increases when $G(V, E)$ comprised a chain of $\text{LGD} + 1$ nodes, and $n - (\text{LGD} + 1)$ fully discon-

**Table 3** NLCO of $G(V, E)$ with total $n$ nodes and $n_f$ fully connected nodes

| $G(V, E)$ | | | NLCO | |
|---|---|---|---|---|
| | | | Value/range | $n \to \infty$ |
| $n = 1$ | | | 0 | |
| $n = 2$ | $n_f = n$ | | 1 | |
| | $n_f = 0$ | | 0 | |
| $n \geq 3, \quad n_f \neq n - 1$ | $n_f > 0$ | | 1 | |
| | $n_f = 0$ | $\text{LGD} = n - 1, \quad n \neq 3$ | 1 | |
| | | $1 \leq \text{LGD} \leq n - 2$ | $\left[\frac{\text{LGD}}{n-1}, 1\right]$ | (0, 1] |
| | | $\text{LGD} = 0$ | 0 | |

nected nodes. In this case, $\text{NEMS} = \frac{\text{LGD}+1}{n(n-1)}$. Since LGD is bounded by $n - 2$, $\lim_{n\to\infty} \frac{\text{LGD}+1}{n(n-1)} = 0$. This result is intuitively correct. Fully disconnected nodes have the lowest diffusion speed. For a component, a chain structure has the lowest diffusion speed. $G(V, E)$ has the lowest diffusion speed if it comprised a chain structure (with minimum component size LGD + 1) and other fully disconnected nodes.

NEMS approaches one as $n$ increases when $G(V, E)$ comprised a clique of $n - 1$ nodes and a fully disconnected node. In this case, $\text{NEMS} = \frac{n-2}{n}$ and $\lim_{n\to\infty} \frac{n-2}{n} = 1$. Since a clique has the highest diffusion speed and $G(V, E)$ does not have any fully connected node, $G(V, E)$ has the highest diffusion speed if it comprised a clique with the maximum order $n - 1$, and a single fully disconnected node;

(d) If $G(V, E)$ does not have any edge (i.e., $\text{LGD} = 0$), $\text{NEMS} = 0$; and
(e) Compared to the NEMS ($0 < \text{NEMS} \leq \frac{1}{3}$) of $G(V, E)$ that comprised a chain, the NEMS ($\text{NEMS} = 0.5$ or 1) of $G(V, E)$ that has at least one fully connected node is higher. When $G(V, E)$ represents the typical structure of most real-world systems ($n_f = 0$ and $1 \leq \text{LGD} \leq n - 2$), $0 < \text{NEMS} < 1$ and NEMS is a useful performance metric to describe how fast entities diffuse in $G(V, E)$.

### Diffusion scale: normalized largest component order

Suppose $v_i$ is the source node that diffuses entities to other nodes in $G(V, E)$. $n_{v_i} - 1$ is the number of nodes that receive entities. Suppose $\forall v_i \in V$ might be the source node, $\max(n_{v_i}) - 1$ is the maximum number of nodes that receive entities if there is only one source node. $\max(n_{v_i})$ is the largest component order in $G(V, E)$. The normalized largest component order (NLCO; Eq. (3)) indicates the maximum scale of (a) terrorist or criminal activities by connected terrorists or criminals; and (b) networked computers, communication devices, and sensors.

$$\text{NLCO} = \frac{\max(n_{v_i}) - 1}{n - 1} \tag{3}$$

**Proposition 3** *Suppose $G(V, E)$ has total n nodes and $n_f$ ($0 \leq n_f \leq n$) fully connected nodes. Table 3 depicts the* NLCO *of $G(V, E)$. $0 \leq \text{NLCO} \leq 1$. $\text{NLCO} \leq c$, $0 \leq c \leq 1$, is a nontrivial property of $G(V, E)$. $\text{NLCO} \leq c$, $c = 0$ or $c = 1$, is a hereditary property of $G(V, E)$. $\text{NLCO} \leq c$, $0 < c < 1$, is a nonhereditary property of $G(V, E)$.*

*Proof of Proposition 3* When $n = 1$, it is defined that $\text{NLCO} = 0$.
When $n = 2$, $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} = \frac{1-1}{2-1} = 0$ if there is no edge; $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} = \frac{2-1}{2-1} = 1$ if there is one edge.
When $n \geq 3$,

(a) $n_f = n$ or $1 \leq n_f \leq n-2$, $G(V, E)$ comprised a single component. $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} = \frac{n-1}{n-1} = 1$;
(b) $n_f \neq n - 1$ (see proof in Proposition 1); and
(c) $n_f = 0$. Three different situations can be analyzed: $\text{LGD} = n - 1$, $1 \leq \text{LGD} \leq n - 2$, and $\text{LGD} = 0$.
   (i) $\text{LGD} = n - 1$. $n \geq 4$ (see proof in Proposition 1). $G(V, E)$ comprised a chain of $n$ nodes. $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} = \frac{n-1}{n-1} = 1$;
   (ii) $1 \leq \text{LGD} \leq n-2$. The minimum largest component order, $\max(n_{v_i})$, is LGD+1. $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} \geq \frac{(\text{LGD}+1)-1}{n-1} = \frac{\text{LGD}}{n-1} \geq \frac{1}{n-1}$. Since $\lim_{n\to\infty} \frac{1}{n-1} = 0$, $\text{NLCO} > 0$. The maximum $max(n_{v_i})$ is $n$ when $G(V, E)$ comprised a single component. Systems with $n_f = 0$ and $1 \leq \text{LGD} \leq n - 2$ may comprise a single component. For example, $G(V, E)$ comprised a chain of $n - 1$ nodes and another node that is connected to one of the nodes on the chain other than the two end nodes. $\text{NLCO} \leq 1$; and
   (iii) $\text{LGD} = 0$. $\text{NLCO} = \frac{\max(n_{v_i})-1}{n-1} = \frac{1-1}{n-1} = 0$.

The desired system property in the recognition problem is $\text{NLCO} \leq c$, where $0 \leq c \leq 1$. This property is nontrivial since it is true for infinitely many networks and false for infinitely many networks for given $c$. NLCO $\leq 1$ is hereditary since the maximum of NEGD is one. NLCO $\leq 0$ if and only if $G(V, E)$ does not have any edge; any subsystem of $G(V, E)$ does not have any edge. NLCO $\leq 0$ is also hereditary. When $0 < c < 1$, NLCO $\leq c$ is nonhereditary. The proof is as follows:

For any value of $c, 0 < c < 1, G(V, E)$ can be constructed such that it includes only one edge and NLCO $= \frac{1}{n-1} \leq c < 1$, $n \geq 3$. A subsystem of $G(V, E)$, which has two nodes and one edge that connects the two nodes, may be produced by removing all nodes without any edge. For this subsystem, NLCO $= 1 > c$. NLCO $\leq c$, $0 < c < 1$, is therefore nonhereditary.

This concludes the proof of Proposition 3. □

Proposition 3 reveals the diffusion scale of $G(V, E)$ ($n \geq 3$):

(a) $G(V, E)$ comprised a single component if $n_f > 0$ or LGD $= n - 1$. $G(V, E)$ may comprise a single component if $n_f = 0$ and LGD $< n - 1$. When $G(V, E)$ comprised a single component, NLCO $= 1$;
(b) If $G(V, E)$ does not have any node that is fully connected (i.e., $n_f = 0$), and does not comprise a chain of $n$ nodes (i.e., $1 \leq$ LGD $\leq n-2$), NLCO is greater than zero and less than or equal to one. NLCO approaches zero as $n$ increases if the LGD of $G(V, E)$ is a constant. For large systems, NLCO $\approx 0$ if LGD $\leq c_1$; $c_1$ is a constant and $c_1 \ll n$; and
(c) If $G(V, E)$ does not have any edge (i.e., LGD $= 0$), NLCO $= \frac{\max(n_{v_i})-1}{n-1} = \frac{1-1}{n-1} = 0$.

## CNI algorithms and complexity

All three performance metrics may be applied to identify critical nodes. For example, the scale and complexity ofterrorist attacks are related to the size of terrorist groups; smaller groups are less likely to launch coordinated large-scale attacks. The NLCO may be used to identify critical terrorists. For another example, to enhance information security, it is important to slow down the spread of viruses in a computer network. The NEMS may be used to identify vulnerable computers. For engineered systems such as the Internet and wireless sensor networks, the significance of using the performance metrics is twofold. First, the performance metrics may be applied to compare different system designs. For instance, the NEMS may be used to gauge the speed of information diffusion in a wireless sensor network, and help select the best network designs. Second, the performance metrics may be applied to identify critical nodes. In a computer network, for instance, the NEGD may be used to identify critical computers whose removal result in a heterogeneous network, which is vulnerable to targeted attacks [1].

Propositions 1 through 3 summarize the performance of the three metrics over generalized systems, and enable the assessment and comparison of systems of different orders and sizes. Algorithms are needed to apply the three metrics to identify critical nodes.

### Nonpolynomial-time CNI algorithms

Given that a maximum of $m$ nodes may be removed from a system of $n$ nodes ($0 < m \leq n$), Algorithm A uses the NEGD to identify the most critical nodes (i.e., the global optimal solution) in the system.

***Algorithm A:***

Step 1: Let $NEGD_{min} = 1$, the maximum possible value of $NEGD$;
Step 2: Determine sets of nodes which may be removed from the system. There are total $\sum_{k=1}^{m} \binom{n}{k}$ different sets;
Step 3: Calculate the $NEGD$ for the system assuming a set of $k$ ($0 < k \leq m$) nodes are removed;
Step 4: Compare the $NEGD$ to the $NEGD_{min}$; if the former is smaller, it is set to be the new $NEGD_{min}$;
Step 5: Repeat Steps 3 and 4 for all $\sum_{k=1}^{m} \binom{n}{k}$ sets. The set of nodes corresponding to the $NEGD_{min}$ is the set of most critical nodes.

The NEGD in Algorithm A may be replaced with NEMS or NLCO to identify the most critical nodes. Since any performance metric must be calculated for $\sum_{k=1}^{m} \binom{n}{k}$ sets, which cannot be completed in polynomial time, the algorithms that identify the most critical nodes require nonpolynomial time. For certificate-checking, suppose it is required that NEGD $\leq c$, where $c$ is a constant and $c \geq 0$, and a maximum of $m$ nodes may be removed from a system of $n$ nodes ($0 < m \leq n$). It takes $O(m)$ time to verify that $k$ ($0 < k \leq m$) nodes are removed from the system. Calculating NEGD requires $O(n^3)$ time. The certificate-checking of NEGD is in time $O(n^3)$. When $c \geq 1$, NEGD $\leq c$ is always true since the maximum of NEGD is one. CNI cannot be performed if desired system property is NEGD $\leq c$, $c \geq 1$. When $c = 0$, NEGD $\leq c$ is a nontrivial and hereditary property of $G(V, E)$ according to Proposition 1. CNI is therefore NP-complete if desired system property is NEGD $\leq 0$. When $0 < c < 1$, NEGD $\leq c$ is a nontrivial and nonhereditary property of $G(V, E)$. There might exist polynomial-time CNI algorithms if desired system property is NEGD $\leq c$, $0 < c < 1$.

Similarly, calculating NEMS or NLCO requires $O(n^3)$ time. The certificate-checking of NEMS or NLCO is in time $O(n^3)$. According to Propositions 2 and 3, both NEMS $\leq 0$ and NLCO $\leq 0$ are nontrivial and hereditary; both NEMS $\leq c$ and NLCO $\leq c$, $0 < c < 1$, are nontrivial and nonhereditary. CNI is NP-complete if desired system property is NEMS $\leq 0$ or NLCO $\leq 0$. There might exist polynomial-time CNI algorithms if desired system property is NEMS $\leq c$ or NLCO $\leq c$, $0 < c < 1$.

## Polynomial-time CNI algorithms

To reduce complexity, Algorithm B uses the NEGD to identify critical nodes in a system step by step.

These three algorithms are applied to the network of 9/11 terrorists [18] to identify the critical nodes. Each square (Fig. 2) represents a terrorist and each link represents communications between two terrorists. There are total 63 nodes and they form a connected network. Note that for coordinated attacks, there cannot be disconnected components. None of the 63 nodes is fully connected (i.e., $n_f = 0$) and the network is not a chain (i.e., LGD $< n-1 = 62$); this reflects the structure of many criminal organizations and is designed for practical purposes such as secrecy. Figures 3, 4 and 5 show the results of applying Algorithm B to the 9/11 terrorist network. Detailed results are included in Tables 4, 5 and 6 in the Appendix. These results validate the three propositions. All three performance metrics are between zero and one. Figure 5 also shows that NLCO $\leq c$, $0 < c < 1$, is nonhereditary because the NLCO does not decrease monotonically and sometimes increases.

Figures 3 and 4 indicate that both the NEGD and NEMS are close to zero after almost 30 nodes are removed or neutralized, which are approximately 50 % of all terrorists. When NEGD is close to zero, most nodes are disconnected, whereas the diffusion speed is almost zero when NEMS is close to zero. The terrorist network cannot launch any large-scale attacks after about 30 nodes are removed using Algorithm B, which is an efficient, polynomial-time algorithm. Figure 5 shows that the NLCO reaches the minimum after about 30 nodes are removed and then increases slightly before it becomes zero. This indicates that the portion of nodes that form a connected group and may launch terrorist attacks is the smallest after 30 nodes are removed. Removing additional nodes does not further decrease the portion until almost all nodes are removed. Overall, the 9/11 terrorists network is destroyed or neutralized after about 30 critical nodes are removed using Algorithm B.

***Algorithm B:***

Step 1: Let $k = 1$, $k$ is the index of steps. One node is removed from the system at each step;
Step 2: Let $NEGD_{min,k} = 1$, the maximum possible value for $NEGD$;
Step 3: Calculate the $NEGD$ for the system assuming one node is removed;
Step 4: Compare the $NEGD$ to the $NEGD_{min,k}$; if the former is smaller, it is set to be the new $NEGD_{min,k}$;
Step 5: Repeat Steps 3 and 4 for all $n + 1 - k$ nodes;
Step 6: Remove the node corresponding to the $NEGD_{min,k}$;
Step 7: Compare $k$ to $m$,
  If k<m
    (a) Increase k by 1;
    (b) Go to Step 2.
  Else
    Find $NEGD_{min} = min\{NEGD_{min,1}, NEGD_{min,2}, \ldots, NEGD_{min,m}\}$. The set of nodes corresponding to the $NEGD_{min}$ is the set of the most critical nodes.

Algorithm B identifies the local optimal solution at each step and removes the node whose removal minimizes the NEGD. This algorithm does not guarantee the global optimum, which is achieved by the nonpolynomial-time CNI algorithms (e.g., Algorithm A). Two algorithms similar to Algorithm B may be written by replacing the NEGD with NEMS or NLCO. Calculating NEGD, NEMS, or NLCO requires $O(n^3)$ time. At each step, a performance metric is calculated for $n + 1 - k$ different systems. Since there are $m$ steps, the overall complexity of each of the three CNI algorithms with local optima is $O(mn^4)$.

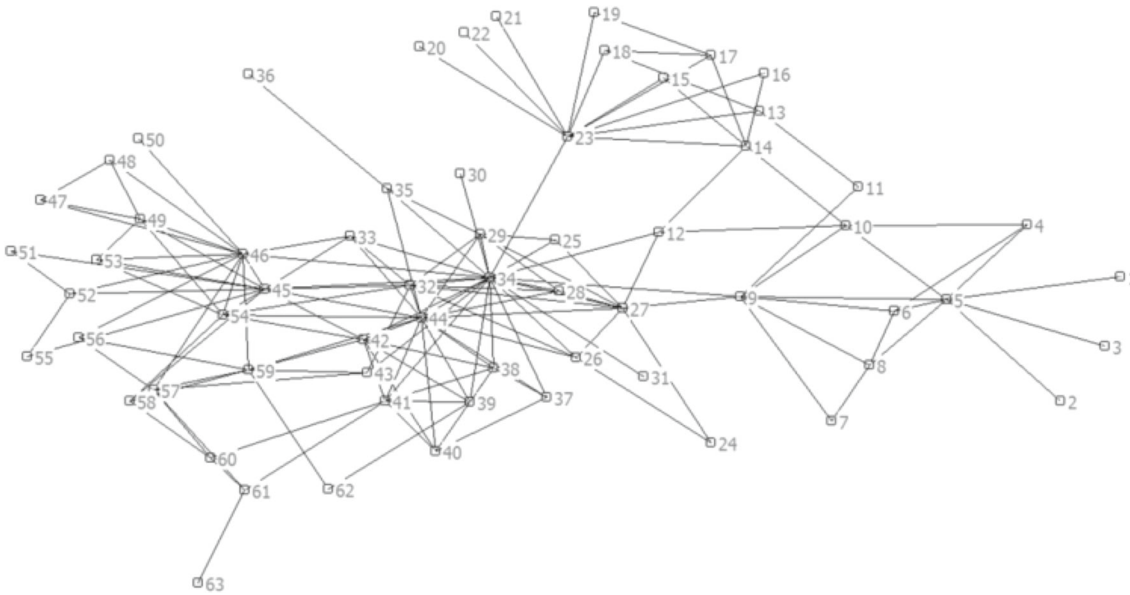



**Fig. 2** September 11 terrorist network (adapted from [18])

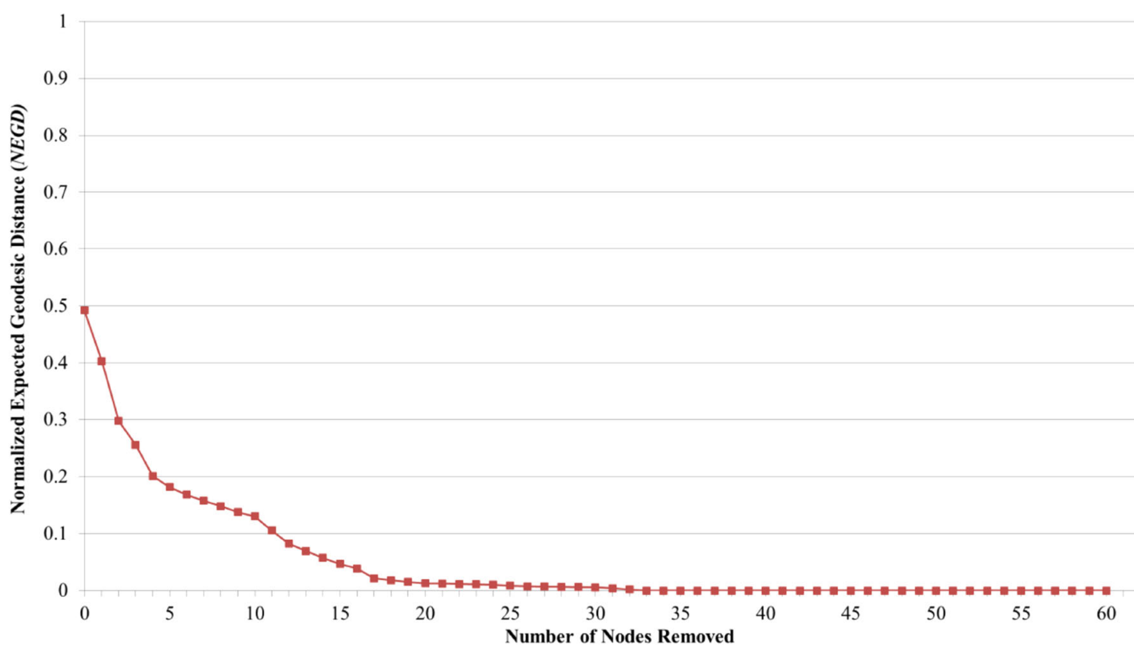



**Fig. 3** NEGD of the 9/11 terrorist network using Algorithm B for CNI

## Comparative analysis of CNI algorithms

To further validate Algorithm B, three CNI algorithms, including degree-based node removal, betweenness-based node removal, and random node removal, are applied to the 9/11 terrorist network. These three algorithms have been widely used to identify critical nodes in complex systems. The degree-based node removal identifies critical nodes based on their degree [1,21]. The node degree, $\theta(v_i)$, is an indicator of $v_i$'s connections with other nodes in a system. Nodes with a larger degree are more critical and are removed or neutralized first. The betweenness-based node removal identifies critical nodes based on their betweenness, which measures the frequency with which a node falls on the shortest paths connecting pairs of other nodes [17]. Betweenness indicates the potential of a node in controlling communications in a system. Nodes with a larger betweenness are more critical and are removed or neutralized first. Equation (4) calculates the betweenness, $Bet(v_k)$, of node $v_k$, where $b_{ij}(v_k)$ (Eq. 5) is the portion of the shortest paths connecting nodes $v_i$ with $v_j$ that contain node $v_k$. $g_{ij}$ in Eq. (5) is the total number of shortest paths connecting $v_i$ and $v_j$ and $g_{ij}(v_k)$

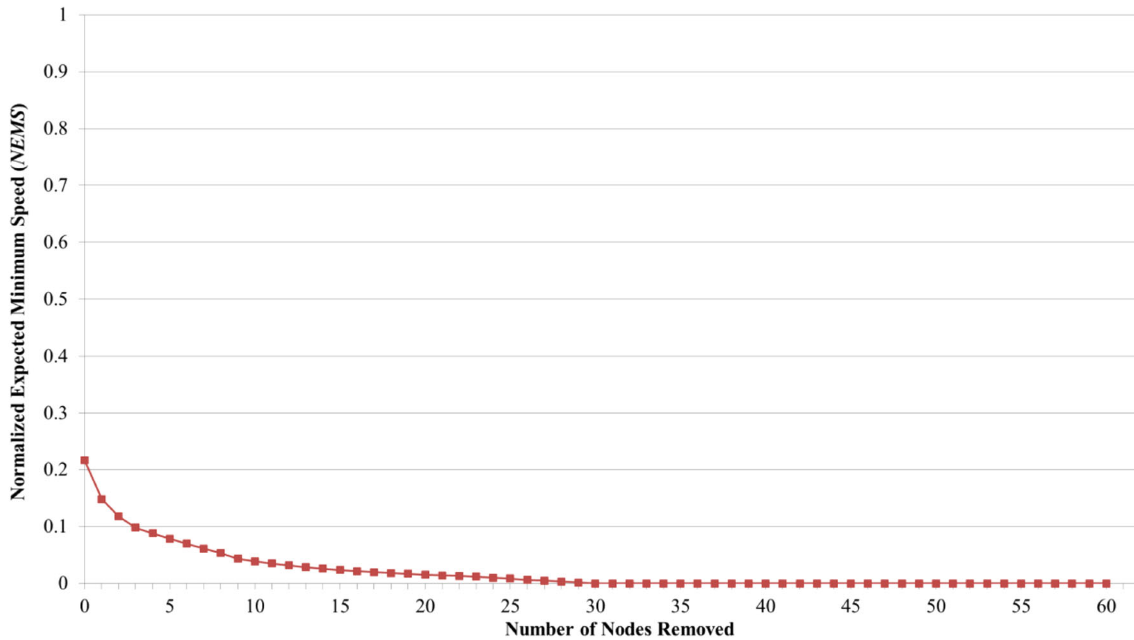



**Fig. 4** NEMS of the 9/11 terrorist network using Algorithm B for CNI

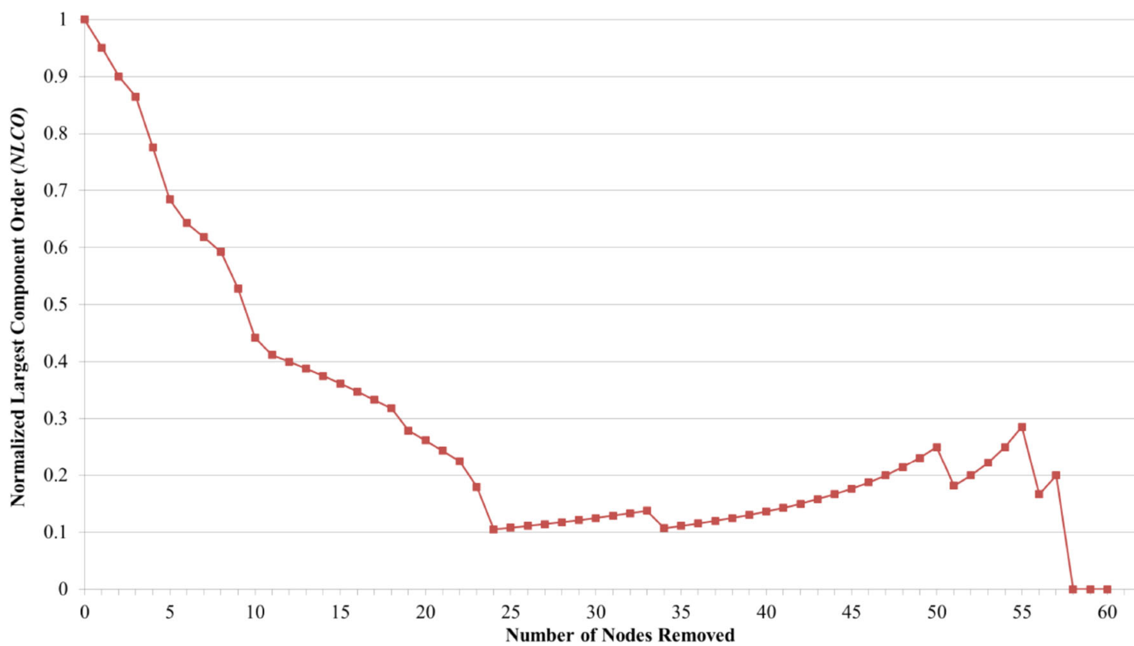



**Fig. 5** NLCO of the 9/11 terrorist network using Algorithm B for CNI

is the number of shortest paths that connect $v_i$ and $v_j$ and contain $v_k$.

$$\mathrm{Bet}(v_k) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} b_{ij}(v_k) \tag{4}$$

$$b_{ij}(v_k) = \frac{1}{g_{ij}} \times g_{ij}(v_k) \tag{5}$$

The random node removal randomly selects a node as the most critical node and removes it from the system. The random node removal is expected to have the worst performance in minimizing the three performance metrics. Figures 6, 7, and 8 compare the performance of the four CNI algorithms, Algorithm B, degree-based node removal, betweenness-based node removal, and random node removal, which are applied to the 9/11 terrorist network. Figure 6 shows that Algorithm B decreases NEGD faster than degree-based node removal and random node removal. The betweenness-based node removal has smaller NEGD compared to Algorithm B between 2 and 24 nodes are removed (Table 4 in the Appen-

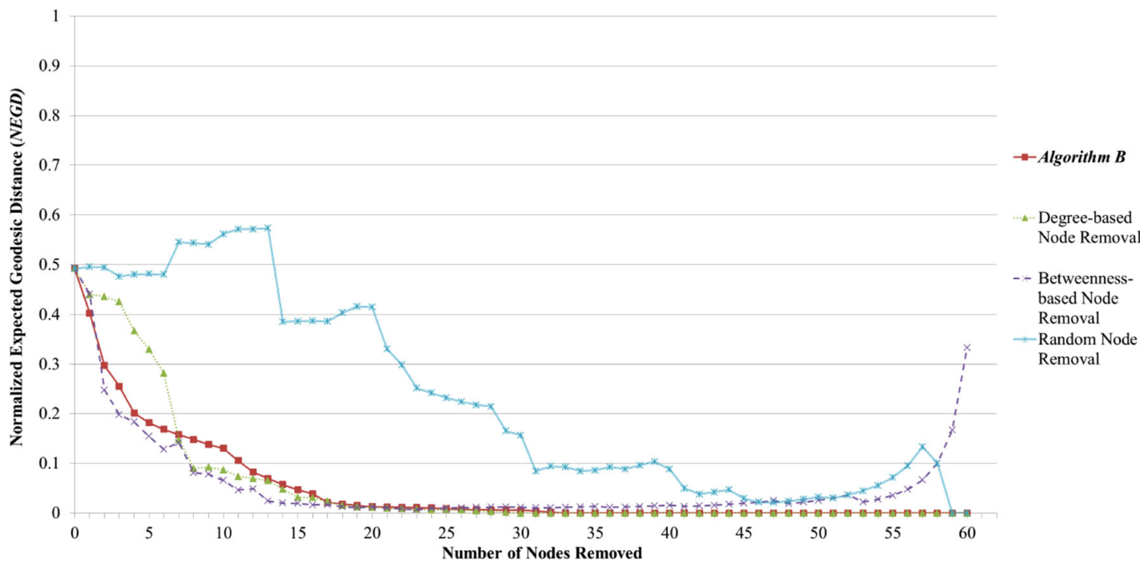



**Fig. 6** Comparison of NEGD with different CNI algorithms (Table 4 in the Appendix)

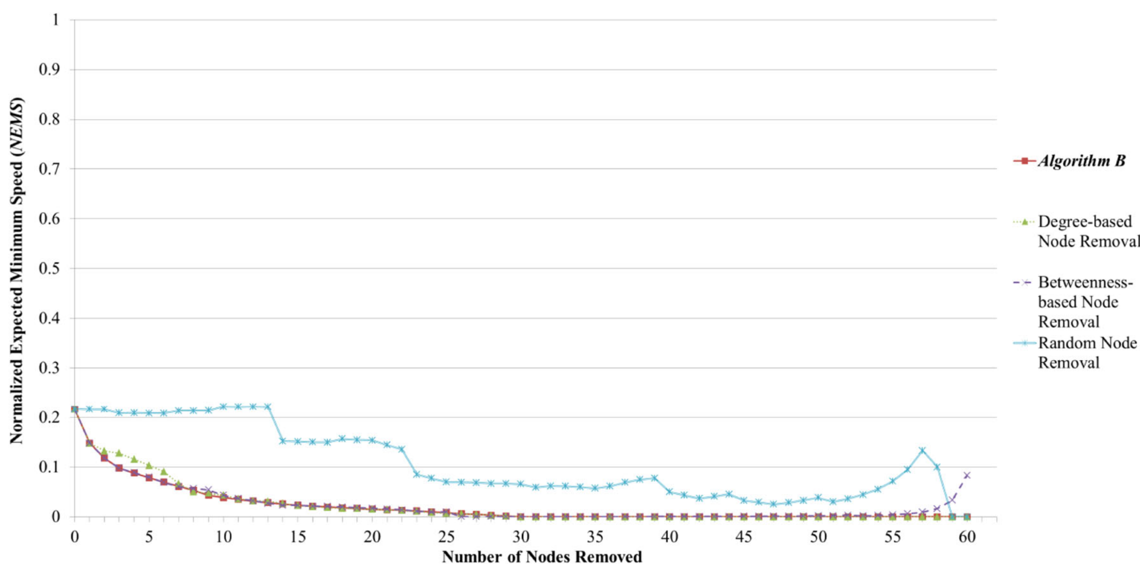



**Fig. 7** Comparison of NEMS with different CNI algorithms (Table 5 in the Appendix)

dix). Starting from the 25th node, however, Algorithm B performs better than the betweenness-based node removal with smaller NEGD. Overall, Algorithm B has the best performance among all four CNI algorithms and consistently decreases NEGD.

Figure 7 shows a similar trend observed in Fig. 6. Algorithm B performs the best in minimizing NEMS. Figure 8 shows that Algorithm B decreases NLCO faster than the random node removal. The degree-based node removal and betweenness-based node removal have smaller NLCO than Algorithm B at the beginning, but larger NLCO when there are a few nodes left in the system. Overall, Algorithm B should be used to minimize the three performance metrics, whereas degree-based node removal and betweenness-based node removal may be used to decrease NLCO more efficiently.

As discussed in Sect. "CNI algorithms and complexity", the computational complexity of Algorithm B is $O(mn^4)$, where $m$ is the number of critical nodes and $n$ is the total number of nodes in a system. The computational complexity of the degree of a node is $O(n)$. To compute the degree of all $n$ nodes, the complexity is $O(n^2)$. The complexity of the degree-based node removal is therefore $O(mn^4)$, which is the same as that of Algorithm B. Using Brandes' Algorithm [6], calculating the betweenness of a node requires $O(ns)$ time, where $s$ denotes the number of edges in a system. Since there are at most $\frac{n(n-1)}{2}$ edges, the complexity of calculating betweenness is $O(n^3)$. The complexity of the betweenness-based node removal is $O(mn^5)$. The betweenness-based node

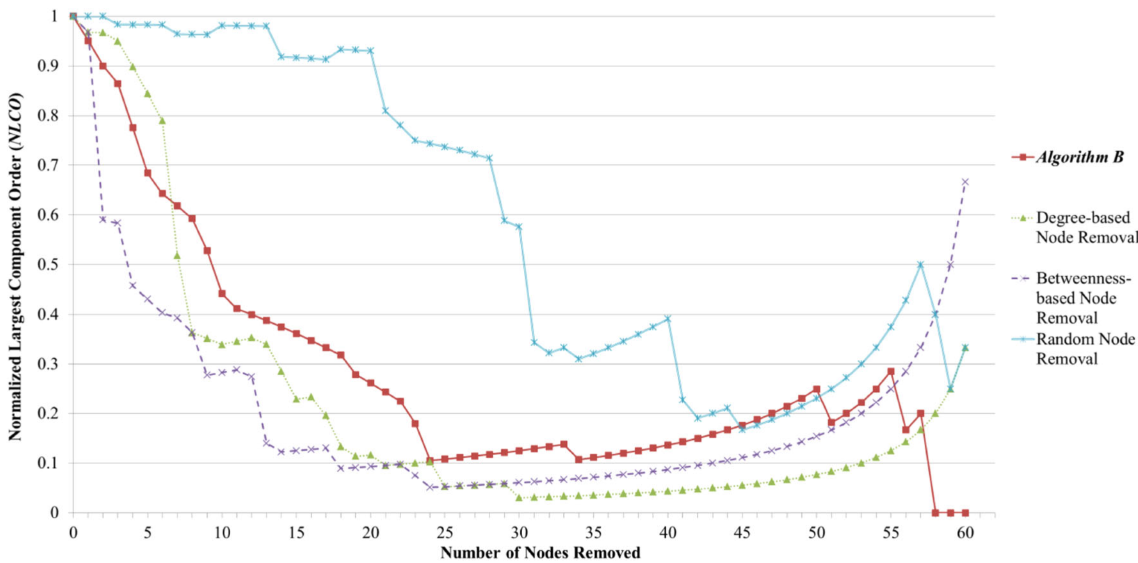



**Fig. 8** Comparison of NLCO with different CNI algorithms (Table 6 in the Appendix)

removal is the second best algorithm among the four CNI algorithms, but requires more computation time compared to Algorithm B. The complexity of the random node removal is $O(mn)$ because the complexity of removing a randomly selected node at each step is $O(n)$.

## Conclusions and future research

Three new performance metrics, NEGD, NEMS, and NLCO, are designed to assess a system's ability of diffusing entities such as information, goods, or diseases. All three metrics are normalized and are between zero and one. The higher their value is, the more capable is a system to diffuse entities. Characteristics of the three metrics are analyzed for generalized systems. All three metrics are nontrivial; they are nonhereditary except for extreme cases (e.g., NEGD $\leq 1$ or NEGD $\leq 0$).

These three performance metrics may be used to identify critical nodes in complex systems. Three nonpolynomial algorithms (Sect. "Nonpolynomial-time CNI algorithms") use the three metrics to identify the most critical nodes (i.e., global optimum). CNI is NP-complete if any of the three metrics is required to be less than or equal to zero. There might exist polynomial-time CNI algorithms if any of the three metrics is required to be less than or equal to a constant that is between but exclusive of zero and one. In Sect. "Polynomial-time CNI algorithms", three polynomial-time algorithms are designed to identify critical nodes step by step (i.e., local optima). These three algorithms with local optima do not guarantee the identification of the global optimum, but their algorithm complexity is $O(mn^4)$, which is in class P, where $m$ is the number of critical nodes to be identified and $n$ is the number of nodes in the system (i.e., system order; $m \leq n$). These polynomial-time algorithms are compared to three other widely used CNI algorithms, including degree-based node removal, betweenness-based node removal, and random node removal. The polynomial-time algorithms developed in this article have the best performance.

CNI is important in controlling complex systems with limited resources. Future research may focus on three areas:

(a) Study the relationship between the three performance metrics and determine whether they can be integrated or additional metrics are needed to assess desired properties of complex systems;
(b) Apply the six algorithms to other real-world complex systems to further validate and compare their performance and complexity;
(c) Design other exact or heuristic optimization algorithms to identify critical nodes. Since the three performance metrics are nontrivial but nonhereditary properties in most cases, there might exist exact optimization algorithms that belong to class P; and
(d) Integrate the DNA and the performance metrics and algorithms developed in this research, and apply them to systems whose properties, e.g., topology or structure, change over time.



## Appendix

See Tables 4, 5 and 6.

**Table 4** Comparison of NEGD with different CNI algorithms

| Number of nodes removed | NEGD | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 0 | 0.492063 | 0.492063 | 0.492063 | 0.492063 |
| 1 | 0.403264 | 0.440508 | 0.440508 | 0.495417 |
| 2 | 0.297814 | 0.436248 | 0.248306 | 0.494171 |
| 3 | 0.255618 | 0.425863 | 0.197902 | 0.476083 |
| 4 | 0.201117 | 0.367167 | 0.183285 | 0.480421 |
| 5 | 0.181791 | 0.330174 | 0.154467 | 0.481650 |
| 6 | 0.168603 | 0.282373 | 0.128133 | 0.479845 |
| 7 | 0.157674 | 0.147774 | 0.141558 | 0.545195 |
| 8 | 0.148092 | 0.090348 | 0.081145 | 0.543569 |
| 9 | 0.137965 | 0.092243 | 0.078092 | 0.540741 |
| 10 | 0.130209 | 0.087083 | 0.065856 | 0.561538 |
| 11 | 0.105688 | 0.073823 | 0.046456 | 0.571493 |
| 12 | 0.082241 | 0.069356 | 0.048628 | 0.571137 |
| 13 | 0.069551 | 0.065539 | 0.024163 | 0.573388 |
| 14 | 0.057483 | 0.047862 | 0.020068 | 0.385326 |
| 15 | 0.046809 | 0.031788 | 0.019504 | 0.386525 |
| 16 | 0.038668 | 0.030924 | 0.016420 | 0.387340 |
| 17 | 0.021643 | 0.023465 | 0.017069 | 0.386197 |
| 18 | 0.018182 | 0.015354 | 0.012795 | 0.403752 |
| 19 | 0.015222 | 0.012156 | 0.010923 | 0.416641 |
| 20 | 0.012625 | 0.011628 | 0.012182 | 0.415124 |
| 21 | 0.012195 | 0.0100658 | 0.010453 | 0.331010 |
| 22 | 0.011585 | 0.00894309 | 0.008537 | 0.298984 |
| 23 | 0.010897 | 0.00769231 | 0.007051 | 0.252137 |
| 24 | 0.010122 | 0.0062978 | 0.009447 | 0.241970 |
| 25 | 0.008535 | 0.00711238 | 0.009957 | 0.232639 |
| 26 | 0.007508 | 0.00600601 | 0.010511 | 0.224543 |
| 27 | 0.007143 | 0.0047619 | 0.011111 | 0.217749 |
| 28 | 0.006723 | 0.00336134 | 0.011765 | 0.214515 |
| 29 | 0.006239 | 0.00178253 | 0.012478 | 0.165107 |
| 30 | 0.005682 | 0 | 0.011364 | 0.156487 |
| 31 | 0.004032 | 0 | 0.010081 | 0.084274 |
| 32 | 0.002151 | 0 | 0.010753 | 0.094086 |
| 33 | 0 | 0 | 0.011494 | 0.092529 |
| 34 | 0 | 0 | 0.012315 | 0.084360 |
| 35 | 0 | 0 | 0.013228 | 0.085979 |
| 36 | 0 | 0 | 0.011396 | 0.092593 |
| 37 | 0 | 0 | 0.012308 | 0.088462 |
| 38 | 0 | 0 | 0.013333 | 0.095833 |
| 39 | 0 | 0 | 0.014493 | 0.103261 |
| 40 | 0 | 0 | 0.015810 | 0.087945 |
| 41 | 0 | 0 | 0.012987 | 0.049784 |
| 42 | 0 | 0 | 0.014286 | 0.038095 |
| 43 | 0 | 0 | 0.015790 | 0.042105 |
| 44 | 0 | 0 | 0.017544 | 0.046784 |
| 45 | 0 | 0 | 0.019608 | 0.029412 |

**Table 4** continued

| Number of nodes removed | NEGD | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 46 | 0 | 0 | 0.022059 | 0.022059 |
| 47 | 0 | 0 | 0.025000 | 0.020833 |
| 48 | 0 | 0 | 0.019048 | 0.023810 |
| 49 | 0 | 0 | 0.021978 | 0.027473 |
| 50 | 0 | 0 | 0.025641 | 0.032051 |
| 51 | 0 | 0 | 0.030303 | 0.030303 |
| 52 | 0 | 0 | 0.036364 | 0.036364 |
| 53 | 0 | 0 | 0.022222 | 0.044444 |
| 54 | 0 | 0 | 0.027778 | 0.055556 |
| 55 | 0 | 0 | 0.035714 | 0.071429 |
| 56 | 0 | 0 | 0.047619 | 0.095238 |
| 57 | 0 | 0 | 0.066667 | 0.133333 |
| 58 | 0 | 0 | 0.100000 | 0.100000 |
| 59 | 0 | 0 | 0.166667 | 0 |
| 60 | 0 | 0 | 0.333333 | 0 |

**Table 5** Comparison of NEMS with different CNI algorithms

| Number of nodes removed | NEMS | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 0 | 0.216931 | 0.216931 | 0.216931 | 0.216931 |
| 1 | 0.147961 | 0.147961 | 0.147961 | 0.216398 |
| 2 | 0.117892 | 0.132310 | 0.117892 | 0.216667 |
| 3 | 0.097892 | 0.127841 | 0.097892 | 0.209171 |
| 4 | 0.088012 | 0.115838 | 0.089022 | 0.209322 |
| 5 | 0.078507 | 0.103247 | 0.079653 | 0.208631 |
| 6 | 0.069737 | 0.090989 | 0.068912 | 0.208777 |
| 7 | 0.061254 | 0.066786 | 0.060758 | 0.213377 |
| 8 | 0.053335 | 0.050595 | 0.057205 | 0.213603 |
| 9 | 0.043528 | 0.047391 | 0.054158 | 0.213836 |
| 10 | 0.038818 | 0.043581 | 0.043541 | 0.222061 |
| 11 | 0.035168 | 0.036295 | 0.036890 | 0.221531 |
| 12 | 0.031909 | 0.032383 | 0.032216 | 0.221941 |
| 13 | 0.028659 | 0.031318 | 0.027075 | 0.221388 |
| 14 | 0.026044 | 0.026680 | 0.023101 | 0.152288 |
| 15 | 0.023569 | 0.022511 | 0.023271 | 0.151348 |
| 16 | 0.021587 | 0.020715 | 0.021508 | 0.150370 |
| 17 | 0.019648 | 0.018479 | 0.021176 | 0.150013 |
| 18 | 0.018014 | 0.016919 | 0.019950 | 0.156831 |
| 19 | 0.016737 | 0.016737 | 0.017970 | 0.154888 |
| 20 | 0.015319 | 0.015319 | 0.017165 | 0.153888 |
| 21 | 0.014131 | 0.014518 | 0.015679 | 0.144541 |
| 22 | 0.013008 | 0.012805 | 0.014024 | 0.135935 |
| 23 | 0.011966 | 0.010897 | 0.011539 | 0.085108 |
| 24 | 0.009897 | 0.008772 | 0.009447 | 0.077553 |
| 25 | 0.008535 | 0.007112 | 0.009957 | 0.069949 |

**Table 5** continued

| Number of nodes removed | NEMS | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 26 | 0.006006 | 0.006006 | 0.000277 | 0.069371 |
| 27 | 0.004762 | 0.004762 | 0.000300 | 0.068554 |
| 28 | 0.003361 | 0.003361 | 0.000327 | 0.067071 |
| 29 | 0.001783 | 0.001783 | 0.000357 | 0.067177 |
| 30 | 0 | 0 | 0.000334 | 0.065956 |
| 31 | 0 | 0 | 0.000305 | 0.059140 |
| 32 | 0 | 0 | 0.000336 | 0.062097 |
| 33 | 0 | 0 | 0.000371 | 0.061782 |
| 34 | 0 | 0 | 0.000411 | 0.060140 |
| 35 | 0 | 0 | 0.000456 | 0.057319 |
| 36 | 0 | 0 | 0.000407 | 0.061728 |
| 37 | 0 | 0 | 0.000456 | 0.068974 |
| 38 | 0 | 0 | 0.000513 | 0.074722 |
| 39 | 0 | 0 | 0.000580 | 0.077597 |
| 40 | 0 | 0 | 0.000659 | 0.050066 |
| 41 | 0 | 0 | 0.000565 | 0.043290 |
| 42 | 0 | 0 | 0.000649 | 0.036905 |
| 43 | 0 | 0 | 0.000752 | 0.040790 |
| 44 | 0 | 0 | 0.000877 | 0.045322 |
| 45 | 0 | 0 | 0.001032 | 0.032680 |
| 46 | 0 | 0 | 0.001225 | 0.029412 |
| 47 | 0 | 0 | 0.001471 | 0.025000 |
| 48 | 0 | 0 | 0.001190 | 0.028571 |
| 49 | 0 | 0 | 0.001465 | 0.032967 |
| 50 | 0 | 0 | 0.001832 | 0.038462 |
| 51 | 0 | 0 | 0.002331 | 0.030303 |
| 52 | 0 | 0 | 0.003030 | 0.036364 |
| 53 | 0 | 0 | 0.002020 | 0.044444 |
| 54 | 0 | 0 | 0.002778 | 0.055556 |
| 55 | 0 | 0 | 0.003968 | 0.071429 |
| 56 | 0 | 0 | 0.005952 | 0.095238 |
| 57 | 0 | 0 | 0.009524 | 0.133333 |
| 58 | 0 | 0 | 0.016667 | 0.100000 |
| 59 | 0 | 0 | 0.033333 | 0 |
| 60 | 0 | 0 | 0.083333 | 0 |

**Table 6** Comparison of NLCO with different CNI algorithms

| Number of nodes removed | NLCO | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0.950820 | 0.967742 | 0.967742 | 1 |
| 2 | 0.900000 | 0.967213 | 0.590164 | 1 |
| 3 | 0.864407 | 0.950000 | 0.583333 | 0.983333 |
| 4 | 0.775862 | 0.898305 | 0.457627 | 0.983051 |
| 5 | 0.684211 | 0.844828 | 0.431034 | 0.982759 |
| 6 | 0.642857 | 0.789474 | 0.403509 | 0.982456 |
| 7 | 0.618182 | 0.517857 | 0.392857 | 0.964286 |
| 8 | 0.592593 | 0.363636 | 0.363636 | 0.963636 |
| 9 | 0.528302 | 0.351852 | 0.277778 | 0.962963 |
| 10 | 0.442308 | 0.339623 | 0.283019 | 0.981132 |
| 11 | 0.411765 | 0.346154 | 0.288462 | 0.980769 |
| 12 | 0.400000 | 0.352941 | 0.274510 | 0.980392 |
| 13 | 0.387755 | 0.340000 | 0.140000 | 0.980000 |
| 14 | 0.375000 | 0.285714 | 0.122449 | 0.918367 |
| 15 | 0.361702 | 0.229167 | 0.125000 | 0.916667 |
| 16 | 0.347826 | 0.234043 | 0.127660 | 0.914894 |
| 17 | 0.333333 | 0.195652 | 0.130435 | 0.913043 |
| 18 | 0.318182 | 0.133333 | 0.088889 | 0.933333 |
| 19 | 0.279070 | 0.113636 | 0.090909 | 0.931818 |
| 20 | 0.261905 | 0.116279 | 0.093023 | 0.930233 |
| 21 | 0.243902 | 0.095238 | 0.095238 | 0.809524 |
| 22 | 0.225000 | 0.097561 | 0.097561 | 0.780488 |
| 23 | 0.179487 | 0.100000 | 0.075000 | 0.750000 |
| 24 | 0.105263 | 0.102564 | 0.051282 | 0.743590 |
| 25 | 0.108108 | 0.052632 | 0.052632 | 0.736842 |
| 26 | 0.111111 | 0.054054 | 0.054054 | 0.729730 |
| 27 | 0.114286 | 0.055556 | 0.055556 | 0.722222 |
| 28 | 0.117647 | 0.057143 | 0.057143 | 0.714286 |
| 29 | 0.121212 | 0.058824 | 0.058824 | 0.588235 |
| 30 | 0.125000 | 0.030303 | 0.060606 | 0.575758 |
| 31 | 0.129032 | 0.031250 | 0.062500 | 0.343750 |
| 32 | 0.133333 | 0.032258 | 0.064516 | 0.322581 |
| 33 | 0.137931 | 0.033333 | 0.066667 | 0.333333 |
| 34 | 0.107143 | 0.034483 | 0.068966 | 0.310345 |
| 35 | 0.111111 | 0.035714 | 0.071429 | 0.321429 |
| 36 | 0.115385 | 0.037037 | 0.074074 | 0.333333 |
| 37 | 0.120000 | 0.038462 | 0.076923 | 0.346154 |
| 38 | 0.125000 | 0.040000 | 0.080000 | 0.360000 |
| 39 | 0.130435 | 0.041667 | 0.083333 | 0.375000 |
| 40 | 0.136364 | 0.043478 | 0.086957 | 0.391304 |
| 41 | 0.142857 | 0.045455 | 0.090909 | 0.227273 |
| 42 | 0.150000 | 0.047619 | 0.095238 | 0.190476 |
| 43 | 0.157895 | 0.050000 | 0.100000 | 0.200000 |
| 44 | 0.166667 | 0.052632 | 0.105263 | 0.210526 |
| 45 | 0.176471 | 0.055556 | 0.111111 | 0.166667 |

**Table 6** continued

| Number of nodes removed | NLCO | | | |
|---|---|---|---|---|
| | Algorithm B | Degree-based node removal | Betweenness-based node removal | Random node removal |
| 46 | 0.187500 | 0.058824 | 0.117647 | 0.176471 |
| 47 | 0.200000 | 0.062500 | 0.125000 | 0.187500 |
| 48 | 0.214286 | 0.066667 | 0.133333 | 0.200000 |
| 49 | 0.230769 | 0.071429 | 0.142857 | 0.214286 |
| 50 | 0.250000 | 0.076923 | 0.153846 | 0.230769 |
| 51 | 0.181818 | 0.083333 | 0.166667 | 0.250000 |
| 52 | 0.200000 | 0.090909 | 0.181818 | 0.272727 |
| 53 | 0.222222 | 0.100000 | 0.200000 | 0.300000 |
| 54 | 0.250000 | 0.111111 | 0.222222 | 0.333333 |
| 55 | 0.285714 | 0.125000 | 0.250000 | 0.375000 |
| 56 | 0.166667 | 0.142857 | 0.285714 | 0.428571 |
| 57 | 0.200000 | 0.166667 | 0.333333 | 0.500000 |
| 58 | 0 | 0.200000 | 0.400000 | 0.400000 |
| 59 | 0 | 0.250000 | 0.500000 | 0.250000 |
| 60 | 0 | 0.333333 | 0.666667 | 0.333333 |

## References


1. Albert R, Jeong H, Barabási A-L (2000) Error and attack tolerance of complex networks. Nature 406:378–382
2. Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286:509–512
3. Bienenstock EJ, Bonacich P (2003) Balancing efficiency and vulnerability in social networks. In: Breiger R, Carley KM, Pattison P (eds) Dynamic social network modeling and analysis: workshop summary and papers. National Research Council, Washington
4. Breiger R, Carley KM, Pattison P (eds) (2003) Dynamic social network modeling and analysis: workshop summary and papers. Committee on human factors, National Research Council, Washington
5. Borgatti SP (2006) Identifying sets of key players in a social network. Comput Math Organ Theory 12:21–34
6. Brandes U (2001) A faster algorithm for betweenness centrality. J Math Soc 25(2):163–177
7. Broder A, Kumar R, Maghoul F, Raghavan P, Rajagopalan S, Stata R, Tomkins A, Wiener J (2000) Graph structure in the Web. Comput Netw 33:309–320
8. Callaway DS, Newman MEJ, Strogatz SH, Watts DJ (2000) Network robustness and fragility: percolation on random graphs. Phys Rev Lett 85(25):5468–5471
9. Carley KM (2006) A dynamic network approach to the assessment of terrorist groups and the impact of alternative courses of action. In: the Proceedings of the visualising network information, RTO-MP-IST-063, Neuilly-sur-Seine, France
10. Chen XW, Nof SY (2007) Prognostics and diagnostics of conflicts and errors over e-work networks. In: Proceedings of the 19th international conference on production research, Chile
11. Chen XW, Nof SY (2010) A decentralized conflict and error detection and prediction model. Int J Prod Res 48(16):4829–4843
12. Chen XW, Landry SJ, Nof SY (2011) A framework of enroute air traffic conflict detection and resolution through complex network analysis. Comput Ind 62:787–794
13. Chen XW, Nof SY (2012) Conflict and error prevention and detection in complex networks. Automatica 48:770–778
14. Cornwell B (2005) A complement-derived centrality index for disconnected graphs. Connections 26(2):70–81
15. Lazer D (2003) Information and innovation in a networked world. In: Breiger R, Carley KM, Pattison P (eds) Dynamic social network modeling and analysis: workshop summary and papers. National Research Council, Washington
16. Dinh TN, Xuan Y, Thai MT, Pardalos PM, Znati T (2012) On new approaches of assessing network vulnerability: hardness and approximation. IEEE/ACM Trans Netw 20(2):609–619
17. Freeman LC (1978) Centrality in social network conceptual clarification. Soc Netw 1:215–239
18. Krebs V (2002) Mapping networks of terrorist cells. Connect 24(3):43–52
19. Lewis JM, Yannakakis M (1980) The node-deletion problem for hereditary properties is NP-complete. J Comput Syst Sci 20:219–230
20. Moon I-C, Carley KM (2007) Modeling and simulation of terrorist networks in social and geospatial dimensions. IEEE Intell Syst Spec Issue Soc Comput 22(5):40–49
21. Newman M (2010) Networks: an introduction. Oxford University Press, Oxford
22. Pero M, Rossi T, Noé C, Sianesi A (2010) An exploratory study of the relation between supply chain topological features and supply chain performance. Int J Prod Econ 123(2):266–278
23. Shen S, Smith JC (2012) Polynomial-time algorithms for solving a class of critical node problems on trees and series-parallel graphs. Networks 60(2):103–119
24. Tsvetovat M, Carley KM (2002) Knowing the enemy: a simulation of terrorist organizations and counter-terrorism strategies. In: the Proceedings of the CASOS conference, Pittsburgh
25. Yannakakis M (1978) Node- and edge-deletion NP-complete problems. In: the Proceedings of the 10th Annual ACM symposium on theory of computing pp 253–264

26. Yin S, Ding SX, Xie X, Luo H (2014) A review on basic data-driven approaches for industrial process monitoring. IEEE Trans Ind Electron 61(11):6418–6428
27. Yin S, Li X, Gao H, Kaynak O (2015) Data-based techniques focused on modern industry: an overview. IEEE Trans Ind Electron 62(1):657–667
28. Zhang C, Ramirez-Marquez JE, Rocco Sanseverino CM (2011) A holistic method for reliability performance assessment and critical components detection in complex networks. IIE Trans 43(9):661–675
29. Zhao Y-P, He P, Nik HS, Ren J (2015) Robust adaptive synchronization of uncertain complex networks with multiple time-varying coupled delays. Complexity 20(6):62–73