Advances in Complex Systems

Multi-hop generalized core percolation on complex networks

Yilun Shang
Department of Computer and Information Sciences, Northumbria University, Newcastle NE1 8ST, United Kingdom
yilun.shang@northumbria.ac.uk

Received (received date)
Revised (revised date)

Recent theoretical study on network robustness has focused primarily on attacks by random selection and global vision, but numerous real-life networks suffer from proximity-based breakdown. Here we introduce the multi-hop generalized core percolation on complex networks, where nodes with degree less than k and their neighbors within L-hop distance are removed progressively from the network. The resulting subgraph is referred to as $G(k, L)$-core, extending the recently proposed Gk-core and classical core of a network. We develop analytical frameworks based upon generating function formalism and rate equation method, showing for instance continuous phase transition for $G(2, 1)$-core and discontinuous phase transition for $G(k, L)$-core with any other combination of k and L. We test our theoretical results on synthetic homogeneous and heterogeneous networks, as well as on a selection of large-scale real-world networks. This unravels, for example, a unique crossover phenomenon rooted in heterogeneous networks, which raises a caution that endeavor to promote network-level robustness could backfire when multi-hop tracing is involved.

Keywords: phase transition; random network; generating function; rate equation; core.

1. Introduction

Robustness of networked systems has been a vigorous research subject in network science over the last two decades, informing the study of various applicable fields such as social and biological networks, infrastructure networks, and technological networks [1, 2]. Most of the early work on network robustness has focused on random attacks and targeted attacks which often require global knowledge of the network such as the degree sequence [3–6]. In reality, many attacks rely on proximity-based tracking, which turns out to be economically cost-effective and practically feasible. Natural examples are the epidemic transmission in contact networks [7] and the effect of floods and earthquakes spreading from epicenters to neighboring areas [8]. Technical examples include cyberspace defense where bots are detected and traced following communication links, and email scams spoofing all users in the address book of a compromised account [9, 10]. Percolation on interdependent networks [11–14], where the deletion of a node in one layer would cause deletion in another layer through dependency links between different layers of networks, has been pro-

posed to interpret the cascading failure in interdependent networks. Moreover, a proximity-based percolation strategy on single networks, referred to as localized attack, characterizes how a network dismantles starting from a seed node, its nearest neighbors, and next nearest neighbors, and so on [15–17].

To quantify resilient network topology, the well-known leaf removal algorithm [18] has been extended recently in the work [19, 20] to the so-call k-leaf removal ($k \geq 2$), in which nodes in a network with degree less than k (k-*leaves*) are recursively removed together with all its nearest neighbors and their incident edges. The resulting subgraph is called the Generalized k-core or Gk-*core* (with $G2$-core being the ordinary *core*). By introducing the parameter k, Gk-core displays a discontinuous phase transition for $k \geq 3$ in contrast to the continuous phase transition in ordinary core percolation, i.e. in the case of $k = 2$ [21–23]. The hierarchy of Gk-cores is a unique metric of network robustness in the sense that the inner cores and their nearest neighbors are well-connected, which is distinct from the graph-theoretic notion "k-core", i.e., the maximal subgraph which contains nodes of degree at least k [24].

Here, motivated by the above work, we extend the Gk-core to $G(k, L)$-core, namely "Generalized (k, L)-core", by introducing a parameter $L \geq 1$ accommodating the multi-hop proximity-based removal strategy. Specifically, we consider a pruning algorithm in which k-leaves are recursively removed from the network together with all its neighbors within L-hop distance in terms of shortest paths in the network. For example, if we choose $L = 1$, then the resulting subgraph $G(k, 1)$-core is equivalent to the Gk-core. This tweak allows us to obtain a refined phase diagram of core percolation. We show that $G(k, L)$-core has a discontinuous phase transition for all combinations of k and L apart from the case of $(k, L) = (2, 1)$. Based upon generating functions and rate equations, we develop theoretical frameworks for $G(k, L)$-core percolation and test them on synthetic homogeneous and heterogeneous networks as well as on a selection of real-world networks. A unique crossover phenomenon for heterogeneous networks is identified, which raises a caveat that endeavor to improve robustness could backfire when multi-hop tracing is involved.

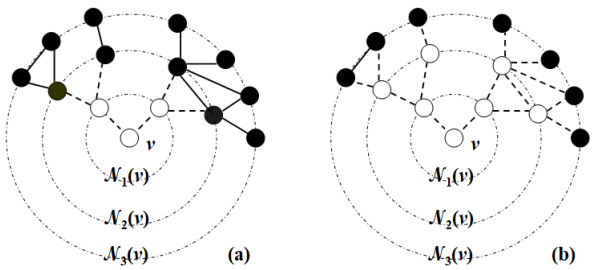

Fig. 1. Schematic illustration of the removal of a 3-leaf node v (a) when $L = 1$ and (b) when $L = 2$. White nodes are removed.

It is worth mentioning that, in addition to network robustness implications, our multi-hop core percolation framework can be linked back to the well-developed applications related to maximum matching and network controllability [25]. The situation of $L \geq 2$ can be mapped to the $L = 1$ case via graph subdivision operation [26]. Therefore, the β-removable nodes (see definition below) as neighbors of a leaf node in $G(2, 2)$-core percolation, for example, are matched in all maximum matching configurations under subdivision. However, this connection only holds in the special case of $k = 2$ and $L = 1$.

2. Theoretical results

2.1. *Generalized (k, L)-core of complex networks*

We consider a random network model with a prescribed degree distribution $P(q)$, which indicates the probability of having q neighbors for a node in the network $\mathcal{G}$. Let $\text{dist}(u, v)$ be the length of the shortest path between two nodes u and v in the network. Denote by $\mathcal{N}_L(v) = \{u : \text{dist}(u, v) = L\}$ the set of L-th nearest neighbors of a node v, and similarly $\overline{\mathcal{N}}_L(v) = \{u : \text{dist}(u, v) \leq L\}$ the set of nodes within L-hop distance of v. Given $k \geq 2$ and $L \geq 1$, our pruning algorithm is straightforward. At each time step, a randomly chosen k-leaf node v (i.e., v has degree less than k) is deleted together with the nodes in $\overline{\mathcal{N}}_L(v)$ and all their incident edges (see Fig. 1). The procedure continues until no k-leaves exist in the remaining network. We call the resulting network the Generalized (k, L)-core, or $G(k, L)$-core for short.

To find the size of the $G(k, L)$-core, we employ the generating function formalism [2, 27] in this section and an alternative rate equation method [29] will be given in Section 2.1. The generating function of the degree distribution is defined by $G_0(x) = \sum_q P(q)x^q$ and $G_1(x) = G_0'(x)G_0'(1)^{-1} = G_0'(x)\langle q\rangle^{-1}$ is the generating function of the excess degree distribution [2, 27], where $\langle q\rangle$ is the average node degree. Using the power property of generating functions, we introduce the generating function for the size distribution of $\mathcal{N}_j(v)$ for a randomly chosen node v as $G^{[j]}(x) = G_0(G_1(\cdots G_1(x)\cdots))$, with $j - 1$ iterations of the function G_1 acting on itself [27]. If we follow a randomly chosen edge to the end node v, the corresponding generating function for the size distribution of $\mathcal{N}_j(v)$ is denoted by $G^{(j)}(x) = G_1(G_1(\cdots G_1(x)\cdots))$, which represents the jth iterate of the function G_1.

Given $k \geq 2$, we divide the nodes into three categories [19, 23]. If a node can become a $(k - 1)$-leaf, it is called α-removable; if a node can lie in $\mathcal{N}_L(v)$ for some k-leaf v, it is called β-removable; if a node cannot be removed and hence belongs to $G(k, L)$-core, it is called non-removable. Let α and β be the probability that a random neighbor of a random node, say u, in a network $\mathcal{G}$ is α-removable and β-removable in $\mathcal{G}\backslash\{u\}$, respectively. Note that the following calculations using generating function formalism are a special case of the cavity method [28], where the cavity refers to the theoretical removal of a link or node when deriving the self-consistent equations.

The network is locally treelike as the probability of having a fixed closed loop goes as n^{-1}, where n is the size of the network, and is thus negligible in the large network limit. Moreover, the above node u lies in $G(k,L)$-core if it has at least $k-1$ out-going nearest neighbors in the $G(k,L)$-core and no node in $\overline{\mathcal{N}}_L(u)$ is α-removable. This means the probabilities satisfy the following self-consistency equation

$$1-\alpha-\beta=\sum_{s_1\geq k-1}\sum_{s_2}\cdots\sum_{s_L}\Bigg[\prod_{j=1}^{L}\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\Big|_{x=0} \cdot\Bigg(\sum_{s=k-1}^{s_1}\binom{s_1}{s}(1-\alpha-\beta)^s\beta^{s_1-s}\Bigg)\cdot(1-\alpha)^{s_2+\cdots+s_L}\Bigg], \tag{1}$$

where s_j is the number of nodes in $\mathcal{N}_j(u)$, namely, the j-th nearest neighbors of u, $\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\big|_{x=0}$ is the probability that u has precisely s_j j-th nearest neighbors (using the derivative property of generating functions [2]), the combinatorial number $\binom{s_1}{s}$ counts the choices of s non-removable neighbors among s_1 neighbors in $\mathcal{N}_1(u)$, and $(1-\alpha)^{s_2+\cdots+s_L}$ is the probability that none of the neighbors in $\overline{\mathcal{N}}_L(u)\backslash\mathcal{N}_1(u)$ are α-removable.

Note that for a β-removable node u, the set $\overline{\mathcal{N}}_L(u)$ contains at least one α-removable node. Therefore, we similarly derive the second self-consistency condition

$$1-\beta=\sum_{s_1}\cdots\sum_{s_L}(1-\alpha)^{s_1+\cdots+s_L}\prod_{j=1}^{L}\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\Big|_{x=0}. \tag{2}$$

By tackling the high-order derivatives in (1) and (2), we find the following amenable expressions for the probabilities α and β (see Appendix A)

$$\alpha=\frac{1}{\langle q\rangle}\Bigg(\sum_{s=0}^{k-2}\frac{(1-\alpha-\beta)^s}{s!}\cdot\frac{d^{s+1}G_0(x)}{dx^{s+1}}\Big|_{x=\beta}\Bigg)\cdot\prod_{j=2}^{L}G^{(j)}(1-\alpha) \tag{3}$$

and

$$\beta=1-\frac{G_0'(1-\alpha)}{\langle q\rangle}\prod_{j=2}^{L}G^{(j)}(1-\alpha). \tag{4}$$

For $L=1$, these equations are consistent with the previous results for Gk-core; see [19] .

Next, we can write down the relative size of $G(k,L)$-core, denoted by $n_{(k,L)}$, which is equivalent to the probability that a random node belongs to the $G(k,L)$-core. We find (see Appendix B)

$$n_{(k,L)}=\Bigg[G_0(1-\alpha)-\sum_{s=0}^{k-1}\frac{(1-\alpha-\beta)^s}{s!}\cdot\frac{d^sG_0(x)}{dx^s}\Big|_{x=\beta}\Bigg]\cdot\prod_{j=2}^{L}G^{[j]}(1-\alpha), \tag{5}$$

which coincides with Equation (5) in [19] for the case $L = 1$. Moreover, for a network with n nodes and l edges, the normalized number of edges of $G(k, L)$-core, signified by $l_{(k,L)}$, can be computed as follows

$$l_{(k,L)} = (1 - \alpha - \beta)^2 \frac{l}{n} = (1 - \alpha - \beta)^2 \frac{\langle q \rangle}{2}, \tag{6}$$

where the term $(1-\alpha-\beta)^2$ means the probability that two end nodes of a random edge belong to the $G(k, L)$-core.

2.2. *Rate equations method*

Following [19, 22, 29], we in this section introduce the rate equations to investigate the structure of the evolution under our pruning process. For ease of analysis, we assume only edges are deleted during the pruning. In other words, at each time step all edges incident to a randomly chosen k-leaf node v and to the nodes in $\overline{\mathcal{N}}_L(v)$ are removed. The procedure continues until all k-leaves become isolated nodes, i.e. nodes of degree zero. Clearly, the remaining network apart from the isolated nodes constitutes the $G(k, L)$-core. As observed in [19], this tweak ensures that the network dynamics is self-averaging, which is essential for the derivation of rate equations.

Suppose the initial network has n nodes and l edges. For each step of the pruning algorithm, the re-scaled time increment is set as $\Delta t = n^{-1}$ following [19]. Letting $P(q, t)$ be the degree distribution at time t, we have $P(q, t) = n(q, t)n^{-1}$, where $n(q, t)$ indicates the average number of nodes having degree q at time t. Therefore, $\frac{dP(q,t)}{dt} = \frac{P(q,t+\Delta t)-P(q,t)}{\Delta t} = n(q, t + \Delta t) - n(q, t) := \Delta n(q, t)$ in the large network limit.

Let $\theta(q)$ be the step function such that $\theta(q) = 1$ if $q > 0$ and $\theta(q) = 0$ if $q \leq 0$. For a randomly chosen k-leaf node v, the average number of nodes in $\overline{\mathcal{N}}_L(v)$ can be seen as

$$\begin{aligned} F_L :=& \sum_{s_1} \cdots \sum_{s_L} (s_1 + \cdots + s_L) \frac{\theta(k - s_1)P(s_1, t)}{\sum_{s_1} \theta(k - s_1)P(s_1, t)} \cdot \left(\prod_{j=2}^{L} \frac{d^{s_j} G^{[j]}(x)}{s_j! dx^{s_j}} \bigg|_{x=0} \right) \\ =& \sum_{s_1} \frac{s_1 \theta(k - s_1)P(s_1, t)}{\sum_{s_1} \theta(k - s_1)P(s_1, t)} \cdot \prod_{j=2}^{L} G^{[j]\prime}(1), \end{aligned} \tag{7}$$

where s_j counts the number of j-th nearest neighbors of v, and $\frac{\theta(k-s_1)P(s_1,t)}{\sum_{s_1} \theta(k-s_1)P(s_1,t)}$ is the probability that v has degree s_1 given it is a k-leaf. Bearing this in mind, we obtain the following rate equation for the evolution of nodes from t to $t + \Delta t$:

$$\begin{aligned} \frac{dP(q, t)}{dt} = \Delta n(q, t) =& - \frac{\theta(k - q)P(q, t)}{\sum_q \theta(k - q)P(q, t)} + \delta_q \cdot (1 + F_L) - F_L \frac{qP(q, t)}{\langle q \rangle_t} \\ &+ (F_{L+1} - F_L) \left(\frac{(q + 1)P(q + 1, t)}{\langle q \rangle_t} - \frac{qP(q, t)}{\langle q \rangle_t} \right). \end{aligned} \tag{8}$$

The first term on the right-hand side of (8) is responsible for the contribution of the randomly chosen k-leaf node v to $\Delta n(q, t)$ (namely, v has degree q with

this probability, and $n(q, t+\Delta t)$ decreases by 1). The second term $\delta_q \cdot (1+F_L)$ corresponds to the number of isolated nodes generated, where $\delta_q = 1$ if $q = 0$ and $\delta_q = 0$ otherwise. When edges incident to nodes in $\overline{\mathcal{N}}_L(v)$ are deleted, $n(q, t+\Delta t)$ decreases by F_L with probability $qP(q,t)\langle q\rangle_t^{-1}$ since $qP(q,t)\langle q\rangle_t^{-1}$ is the probability that a node in $\overline{\mathcal{N}}_L(v)$ (following a random edge) has degree q. The last term in (8) corresponds to the contribution of the nodes in $\mathcal{N}_{L+1}(v)$: $n(q, t+\Delta t)$ increases by $F_{L+1} - F_L$ with probability $(q+1)P(q+1,t)\langle q\rangle_t^{-1}$ and decreases by the same amount with probability $qP(q,t)\langle q\rangle_t^{-1}$. By combining all the cases that may alter $n(q, t+\Delta t)$, we arrive at the rate equation (8), which agrees with the case of $L = 1$; see [19] .

Next, we estimate the number of edges, $l(t) - l(t+\Delta t)$, that are deleted on average in each time step, where $l(t)$ indicates the average number of edges at time t. When L is odd, we have

$$l(t) - l(t+\Delta t) = [F_1 + (F_3 - F_2) + \cdots + (F_L - F_{L-1})] \cdot \frac{\langle q^2\rangle_t}{\langle q\rangle_t}, \tag{9}$$

where $\langle q^2\rangle_t \langle q\rangle_t^{-1}$ is the average degree of the end nodes of a randomly chosen edge. Similarly, if L is even, we have

$$\begin{aligned} l(t) - l(t+\Delta t) =& F_1 + [(F_2 - F_1) + (F_4 - F_3) \\ &+ \cdots + (F_L - F_{L-1})] \cdot \frac{\langle q^2\rangle_t}{\langle q\rangle_t}, \end{aligned} \tag{10}$$

where F_1 accounts for the average number of nearest neighbors of a randomly chosen k-leaf. Therefore, the evolution equation for the dynamics of edges from t to $t+\Delta t$ is given by

$$\frac{l'(t)}{n} = \frac{l(t+\Delta t) - l(t)}{n \cdot \Delta t} = -(l(t) - l(t+\Delta t)), \tag{11}$$

which depends on the parity of L. In particular, if $L = 1$, we have $F_1 = \frac{\sum_q q\theta(k-q)P(q,t)}{\sum_q \theta(k-q)P(q,t)}$ by (7). Applying (9) to (11) we recover Equation (10) in [19].

We perform the above modified pruning procedure on a network of n nodes and l edges until some time t^* when all k-leaves become isolated nodes. The relative size and the normalized number of edges of $G(k, L)$ are given, respectively, by

$$n_{(k,L)} = 1 - P(0, t^*) \tag{12}$$

and

$$l_{(k,L)} = \frac{l(t^*)}{n}. \tag{13}$$

In terms of numerical calculation, we can solve the set of differential equation system (8) iteratively given a finite maximum degree $q_{\max}$. The termination time t^* is determined by $P(q, t^*) = 0$ for $q = 1, \cdots, k-1$. Moreover, in view of (9)-(11), $l_{(k,L)}$ is independent of the network size n in the thermodynamic limit (i.e., $n \to \infty$) and can be solved straightforwardly.

3. Synthetic networks

In this section, we conduct numerical simulations along with the analytical results derived in Section 2 for synthetic homogeneous random networks following Poisson degree distributions and heterogeneous random networks with exponential tails. Note that scale-free networks have no core [23] and hence no $G(k, L)$-core. All the simulations are based on networks with $n = 10^7$.

3.1. *Erdős-Rényi networks*

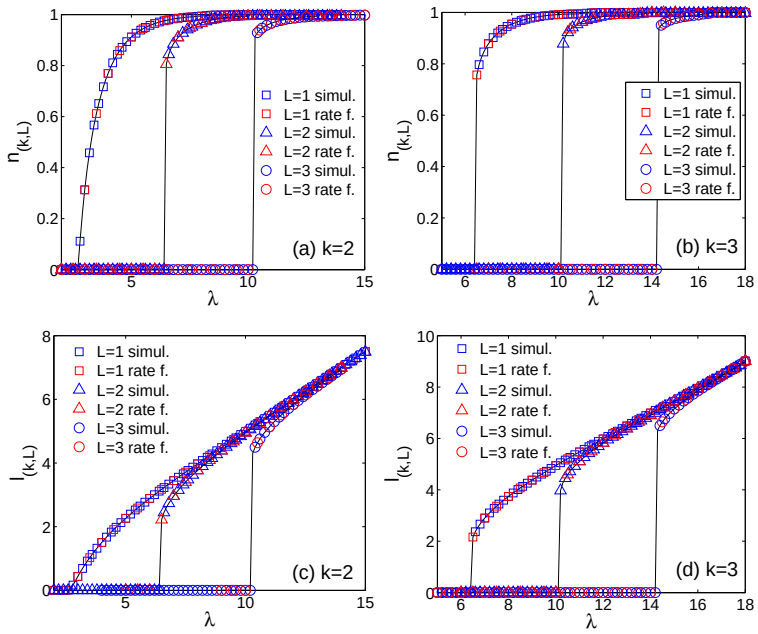

Fig. 2. Top row: fractions of $G(k, L)$-core $n_{(k,L)}$ as functions of λ for $L = 1$ (squares), $L = 2$ (triangles), and $L = 3$ (circles) when (a) $k = 2$ and (b) $k = 3$. Bottom row: normalized edge numbers of $G(k, L)$-core $l_{(k,L)}$ as functions of λ for $L = 1$ (squares), $L = 2$ (triangles), and $L = 3$ (circles) when (c) $k = 2$ and (d) $k = 3$. Curves are theoretical results based on Eqs. (5) and (6); red symbols are obtained from rate equation approach Eqs. (12) and (13); blue symbols correspond to simulations averaged over 30 realizations of ER networks with $n = 10^7$ and average degree λ.

We first consider Erdős-Rényi (ER) random networks with degree distribution $P(q) = e^{-\lambda}\lambda^q/q!$ for $q \geq 0$. The average degree of an ER network is given by $\langle q \rangle = \lambda$.

In Fig. 2 we exhibit the relative size $n_{(k,L)}$ as well as the normalized number of edges $l_{(k,L)}$ for the $G(k, L)$-core for varied degree parameter k and multi-hop parameter L in ER networks. In the calculation of rate equation, we set $q_{\max} = 40$ as $P(q_{\max})$ is less than n^{-1} in all our networks (In fact, the maximum degree of such

an ER network is only around $(\ln\ln n)^{-1}\ln n$ with high probability [31].) Several interesting observations are as follows.

Firstly, numerical simulations, theoretical results based on generating functions and rate equations agree well with each other. Secondly, for $k=2$ and $L=1$ the $G(k,L)$-core shows continuous phase transition for both $n_{(k,L)}$ and $l_{(k,L)}$, while for all other scenarios with larger k or larger L only first-order percolation transition behavior is observed. Our result is in line with [19, 23] for $L=1$, namely, $k=3$ marks the turning point of Gk-core percolation. However, for multi-hop generalized core percolation with $L\geq 2$, such a turning point disappears: the $G(k,L)$-core will emerge abruptly at a critical threshold value $\lambda^*=\lambda^*(k,L)$ for any $k\geq 2$. Thirdly, the phase transitions for $l_{(k,L)}$ is consistent with those for $n_{(k,L)}$. Moreover, the connectivity of $G(k,L)$-core grows nearly linearly with the density of the network. For example, from Fig 2(c) we observe that the average degree of $G(2,2)$-core is around 4 when it first appears at $\lambda\approx 6.5$ and then increases gradually with respect to λ.

To better appreciate the discontinuous phase transition, we show the evolution of λ^* in Fig. 3 for different combinations of k and L. The critical average degree λ^* is seen to increase almost linearly with respect to both parameters, which implies the pruning process unravels the robustness of ER networks in terms of the emergence of core structure gradually and steadily.

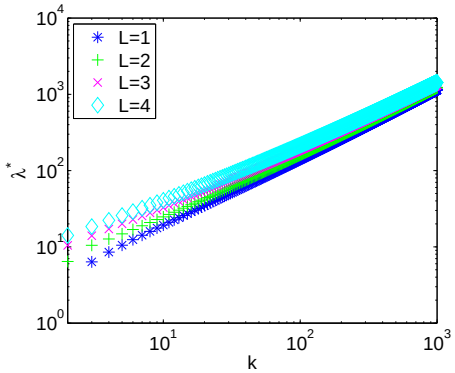

Fig. 3. The critical threshold λ^* for ER networks as a function of k for $L=1$ (stars), $L=2$ (pluses), $L=3$ (crosses), and $L=4$ (diamonds) based on numerical calculation via Eq. (5) where $n_{(k,L)}>10^{-3}$.

3.2. *Exponential networks*

We next consider exponentially distributed networks where degrees follow $P(q)\propto(1+q)^2e^{-q/\gamma}$ for $q\geq 0$, where $\gamma>0$ is the exponent. Exponentially distributed networks are heterogenous with quasi-heavy tails and seen in various real-life networks

[2, 32]. Although scale-free networks always lead to trivial $n_{(k,L)}$, some intriguing phenomena are revealed by such asymptotically exponential networks.

Fig. 4 shows how $n_{(k,L)}$ and $l_{(k,L)}$ evolve with respect to γ for different k and L. Similarly as in the case of ER networks in Fig. 2, continuous phase transition is only observed for $G(2,1)$-core percolation and discontinuous phase transition dominates all the other cases. The critical threshold value for the phase transition shifts toward right indicating that the $G(k,L)$-core for an exponential network is much smaller than that for an ER network with the same density. For example, the $G(2,2)$-core for an ER network with average degree 15 has size $n_{(2,2)} \approx 1$ (see Fig. 2(a)) while $n_{(2,2)}$ for the exponential network with the same average degree is around zero (see Fig. 4(a)). As low-degree nodes (k-leaves) in exponential networks are likely to connect to hubs within their multi-hop neighborhoods, our pruning algorithm causes much more serious damage to such heterogenous networks than homogeneous ER networks.

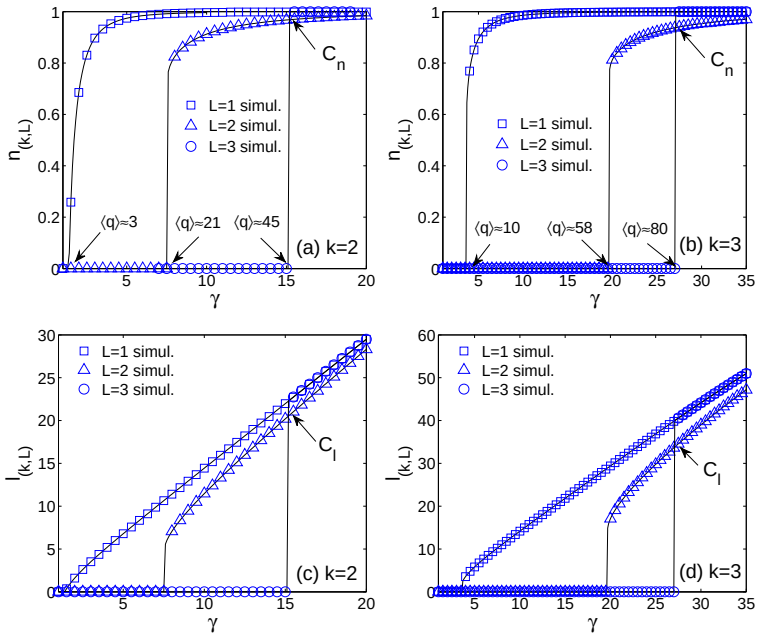

Fig. 4. Top row: fractions of $G(k,L)$-core $n_{(k,L)}$ as functions of γ for $L=1$ (squares), $L=2$ (triangles), and $L=3$ (circles) when (a) $k=2$ and (b) $k=3$. Bottom row: normalized edge numbers of $G(k,L)$-core $l_{(k,L)}$ as functions of γ for $L=1$ (squares), $L=2$ (triangles), and $L=3$ (circles) when (c) $k=2$ and (d) $k=3$. Curves are theoretical results based on Eqs. (5) and (6); symbols correspond to simulations averaged over 30 realizations of exponential networks with $n=10^7$ and exponent γ. The crossover points for $n_{(k,L)}$ and $l_{(k,L)}$ are signified by C_n and C_l, respectively.

Interestingly, we observed from Fig. 4 the appearance of crossover points of $n_{(k,L)}$

(and $l_{(k,L)}$) for $L = 2$ and $L = 3$. For example, Fig. 4(a) and Fig. 4(c) confirm that $G(2,3)$-core is larger and denser than $G(2,2)$-core for networks with $\langle q\rangle \gtrapprox 45$ while $G(2,3)$-core does not exist for $\langle q\rangle \lessapprox 45$. This situation, however, does not exist in ER networks. We contend that the phenomenon that larger multi-hop parameter L could result in larger multi-hop generalized core could find its origin in the network heterogeneity. An example scenario is shown in Fig. 5, where a sparse subnetwork (green network containing k-leaves) is connected to a well-connected subnetwork (orange network). Therefore, $n_{(k,L)}$ and $l_{(k,L)}$ are not monotonic with respect to L in general. This yields an important implication that effort to enhance robustness could backfire — for example, a heterogeneous network may become less robust when facing virus infection (through compromising weak nodes and their multi-hop neighbors) with reduced tracing capability.

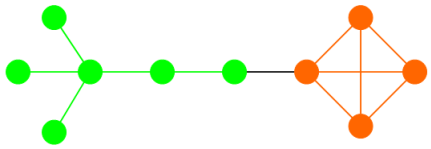

Fig. 5. An example network where $n_{(2,2)} = 0$ but $n_{(2,3)} = 4$. The orange nodes and edges form the $G(2,3)$-core while the green nodes are removed through the $G(2,3)$-core percolation.

4. Real-world networks

We apply the $G(k,L)$-core percolation to a couple of real-world networks in this section. Table 1 presents the relative size and normalized number of edges of $G(k,L)$-core for the Web graph [30] from the technology field, whose nodes are web pages and links are hyperlinks connecting two pages. Web is the undirected version with $n = 163598$ nodes and $l = 16898307$ edges crawled in 2004. The second graph is the Brain network [33], where nodes are human brain neurons and edges are fiber tracts linking neurons. This graph has $n = 750742$ nodes and $l = 175235019$ edges. The results for Brain network are summarized in Table 2.

$n_{(k,L)}/l_{(k,L)}$	Web		
	$k=2$	$k=3$	$k=4$
$L=1$	0.610/8.541	0.498/8.163	0.360/6.372
$L=2$	0.554/5.088	0.347/4.701	0.285/3.036
$L=3$	0.342/4.846	0.168/3.428	0.141/1.965

Table 1. $n_{(k,L)}$ and $l_{(k,L)}$ of $G(k,L)$-core for Web network.

$n_{(k,L)}/l_{(k,L)}$	Brain		
	$k=2$	$k=3$	$k=4$
$L=1$	0.892/18.310	0.653/13.548	0.412/7.345
$L=2$	0.465/12.167	0.326/6.850	0.120/3.464
$L=3$	0.221/9.410	0.104/3.391	0.075/0.901

Table 2. $n_{(k,L)}$ and $l_{(k,L)}$ of $G(k,L)$-core for Brain network.

Both Web and Brain have approximately heavy tails [30, 33], which would result in very small or vanishing core structure according to the theoretical and numerical analysis in Section 3. However, they posses apparently non-trivial $G(k,L)$-cores for $k \leq 4$ and $L \leq 3$, which are substantially larger than what would be expected. Similarly to what was observed in Gk-core percolation and core percolation, we believe wealthy structural features such as correlation, motif or clustering in real networks could play an essential role in the $G(k,L)$-core percolation.

5. Conclusion

In summary, we have introduced the multi-hop generalized core percolation, or $G(k,L)$-core percolation, associating a degree parameter $k \geq 2$ with a multi-hop parameter $L \geq 1$. The proposed pruning procedure extends the Gk-core pruning as well as the k-core pruning by incorporating the L-hop tracing mechanism. For all different combinations of (k,L) except $k=2$ and $L=1$, first-order phase transition is observed in the number of nodes and the number of edges of $G(k,L)$-cores, which offers a new perspective on the network robustness against attacks with tracing capability. Analytical frameworks based on generating function formalism and rate equation method have been established. We tested the theoretical results on synthetic networks as well as real-world networks. The crossover phenomenon revealed in heterogeneous networks highlights a non-monotonicity risk regarding enhancement to network robustness. As a future work, it would be interesting to reveal the topological effect of real-world networks on core structure by randomizing the networks such that the degree sequence is preserved and re-calculating the core.

Acknowledgments

This work was supported by a starting grant of Northumbria University and UoA Flexible Fund Grant No. 201920A1001. The author would like to thank the two anonymous reviewers and the handling editor Márton Karsai for valuable comments that helped improve the paper.

Appendix A: The probabilities α and β

First, note that $(1-\alpha)^{s_1}=\sum_{s=0}^{s_1}\binom{s_1}{s}(1-\alpha-\beta)^s\beta^{s_1-s}$ by the binomial theorem. It follows from (1) and (2) that

$$\begin{aligned}\alpha=&\sum_{s_1}\cdots\sum_{s_L}(1-\alpha)^{s_2+\cdots+s_L}\prod_{j=1}^{L}\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\bigg|_{x=0}\cdot\sum_{s=0}^{k-2}\binom{s_1}{s}(1-\alpha-\beta)^s\beta^{s_1-s}\\ =&\sum_{s=0}^{k-2}\frac{(1-\alpha-\beta)^s}{s!}\bigg[\sum_{s_1}\frac{\beta^{s_1-s}}{(s_1-s)!}\frac{d^{s_1}G_1(x)}{dx^{s_1}}\bigg|_{x=0}\bigg]\\ &\cdot\sum_{s_2}\cdots\sum_{s_L}(1-\alpha)^{s_2+\cdots+s_L}\prod_{j=2}^{L}\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\bigg|_{x=0}.\end{aligned}\tag{14}$$

Drawing on the properties of generating functions, we calculate

$$\frac{d^{s_1}G_1(x)}{dx^{s_1}}\bigg|_{x=0}=\langle q\rangle^{-1}(s_1+1)P(s_1+1),\tag{15}$$

$$\frac{d^{s+1}G_0(x)}{dx^{s+1}}\bigg|_{x=\beta}=\sum_{s_1}\frac{(s_1+1)!}{(s_1-s)!}P(s_1+1)\beta^{s_1-s},\tag{16}$$

and

$$\prod_{j=2}^{L}G^{(j)}(1-\alpha)=\sum_{s_2}\cdots\sum_{s_L}(1-\alpha)^{s_2+\cdots+s_L}\cdot\prod_{j=2}^{L}\frac{d^{s_j}G^{(j)}(x)}{s_j!dx^{s_j}}\bigg|_{x=0}.\tag{17}$$

The expression (3) for the probability α can be obtained by applying (15)-(17) to (14).

Finally, the formula (4) for β can be derived likewise by invoking (2) and $G_0'(1-\alpha)=\sum_q qP(q)(1-\alpha)^{q-1}$.

Appendix B: Relative size of $G(k,L)$-core

To calculate the relative size $n_{(k,L)}$, we note that a randomly chosen node v belongs to $G(k,L)$ if it has at least k nearest neighbors in the $G(k,L)$-core and no node in

$\overline{\mathcal{N}}_L(u)$ is α-removable. Similarly as in (1), we obtain

$$\begin{aligned}
n_{(k,L)} = &\sum_{s_1\geq k}\sum_{s_2}\cdots\sum_{s_L}\Bigg[\prod_{j=1}^{L}\frac{d^{s_j}G^{[j]}(x)}{s_j!dx^{s_j}}\Bigg|_{x=0} \\
&\cdot\left(\sum_{s=k}^{s_1}\binom{s_1}{s}(1-\alpha-\beta)^s\beta^{s_1-s}\right)\cdot(1-\alpha)^{s_2+\cdots+s_L}\Bigg] \\
=&\sum_{s_1}\cdots\sum_{s_L}\Bigg[\prod_{j=1}^{L}\frac{d^{s_j}G^{[j]}(x)}{s_j!dx^{s_j}}\Bigg|_{x=0} \\
&\cdot(1-\alpha)^{s_2+\cdots+s_L}\cdot\left((1-\alpha)^{s_1}-\sum_{s=0}^{k-1}\binom{s_1}{s}(1-\alpha-\beta)^s\beta^{s_1-s}\right)\Bigg] \\
=&\sum_{s_1}\cdots\sum_{s_L}\prod_{j=1}^{L}\frac{d^{s_j}G^{[j]}(x)}{s_j!dx^{s_j}}\Bigg|_{x=0}\cdot(1-\alpha)^{s_1+\cdots+s_L} \\
&-\sum_{s_2}\cdots\sum_{s_L}\prod_{j=2}^{L}\frac{d^{s_j}G^{[j]}(x)}{s_j!dx^{s_j}}\Bigg|_{x=0}\cdot(1-\alpha)^{s_2+\cdots+s_L} \\
&\cdot\left(\sum_{s=0}^{k-1}\frac{(1-\alpha-\beta)^s}{s!}\sum_{s_1}\frac{\beta^{s_1-s}}{(s_1-s)!}\frac{d^{s_1}G^{[1]}(x)}{dx^{s_1}}\Bigg|_{x=0}\right) \\
=&\sum_{s_2}\cdots\sum_{s_L}\prod_{j=2}^{L}\frac{d^{s_j}G^{[j]}(x)}{s_j!dx^{s_j}}\Bigg|_{x=0}\cdot(1-\alpha)^{s_2+\cdots+s_L} \\
&\cdot\left[\sum_{s_1}\frac{d^{s_1}G^{[1]}(x)}{s_1!dx^{s_1}}\Bigg|_{x=0}\cdot(1-\alpha)^{s_1}-\sum_{s=0}^{k-1}\frac{(1-\alpha-\beta)^s}{s!}\frac{d^sG_0(x)}{dx^s}\Bigg|_{x=\beta}\right]. \quad (18)
\end{aligned}$$

Drawing on (17), (18) and the definition $G^{[1]}(x) = G_0(x)$, we readily derive Equation (5).

Conflict of interest disclosure

The author declares that there is no conflict of interest regarding the publication of this article.

References

[1] Cohen, R. and Havlin, S., *Complex Networks: Structure, Robustness and Function*, Cambridge University Press, Cambridge, 2010.

[2] Newman, M. E. J., *Networks: An Introduction*, Oxford University Press, Oxford, 2010.

[3] Albert, R., Jeong, H., Barabási, A.-L., Error and attack tolerance of complex networks, *Nature* **406** (2000) 378.

[4] DeDomenico, M. and Arenas, A., Modeling structure and resilience of the dark network, *Phys. Rev. E* **95** (2017) 022313.

[5] Iyer, S., Killingback, T., Sundaram, B. and Wang, Z., Attack robustness and centrality of complex networks, *PLoS ONE* **8** (2013) e59613.
[6] Shang, Y., Unveiling robustness and heterogeneity through percolation triggered by random-link breakdown, *Phys. Rev. E* **90** (2014) 032820.
[7] Massaro, E., Ganin, A., Perra, N., Linkov, I. and Vespignani, A., Resilience management during large-scale epidemic outbreaks, *Sci. Rep.* **8** (2018) 1859.
[8] Wang, W., Yang, S., Stanley, H. E. and Gao, J., Local floods induce large-scale abrupt failures of road networks, *Nat. Commmun.* **10** (2019) 2114.
[9] Huang, K., Siegel, M. and Madnick, S., Systematically understanding the cyber attack business: a survey, *ACM Comput. Surv.* **51** (2018) 70.
[10] Junger, M., Montoya, L. and Overink, F.-J., Priming and warnings are not effective to prevent social engineering attacks, *Comput. Hum. Behav.* **66** (2017) 75–87.
[11] Aleta, A. and Moreno, Y., Multilayer networks in a nutshell, *Annu. Rev. Condens. Matter Phys.* **10** (2019) 45–62.
[12] Buldyrev, S. V., Parshani, R., Paul, G., Stanley, H. E. and Havlin, S., Catastrophic cascade of failures in interdependent networks, *Nature* **464** (2010) 1025–1028.
[13] Danzier, M. M., Bonamassa, I., Boccaletti, S. and Havlin, S., Dynamic interdependence and competition in multilayer networks, *Nat. Phys.* **15** (2019) 178–185.
[14] Radicchi, F. and Bianconi, G., Redundant interdependencies boost the robustness of multiplex networks, *Phys. Rev. X* **7** (2017) 011013.
[15] Dong, G., Xiao, H., Wang, F., Du, R., Shao, S., Tian, L., Stanley, H. E. and Havlin, S., Localized attack on networks with clustering, *New J. Phys.* **21** (2019) 013014.
[16] Shang, Y., Localized recovery of complex networks against failure, *Sci. Rep.* **6** (2016) 30521.
[17] Shao, S., Huang, X., Stanley, H. E. and Havlin, S., Percolation of localized attack on complex networks, *New J. Phys.* **17** (2015) 023049.
[18] Karp, R. M. and Sipser, M., Maximum matching in sparse random graphs, *Proc. 22nd Annual IEEE Symposium on Foundations of Computer Science*, IEEE, Piscataway, NJ, pp. 364–375, 1981.
[19] Azimi-Tafreshi, N., Osat, S. and Dorogovtsev, S. N., Generalization of core percolation on complex networks, *Phys. Rev. E* **99** (2019) 022312.
[20] Shang, Y., Attack robustness and stability of generalized k-cores, *New J. Phys.* **21** (2019) 093013.
[21] Azimi-Tafreshi, N., Dorogovtsev, S. N. and Mendes, J. F. F., Core organization of directed complex networks, *Phys. Rev. E* **87** (2013) 032815.
[22] Bauer, M. and Golinelli, O., Core percolation in random graphs: a critical phenomena analysis, *Eur. Phys. J. B* **24** (2001) 339–352.
[23] Liu, Y.-Y., Csóka, E, Zhou, H. and Pósfai, M., Core percolation on complex networks, *Phys. Rev. Lett.* **109** (2012) 205703.
[24] Dorogovtsev, S. N., Goltsev, A. V. and Mendes, J. F. F., k-core organization of complex networks, *Phys. Rev. Lett.* **96** (2006) 040601.
[25] Liu, Y.-Y. and Barabási, A.-L., Control principles of complex systems, *Rev. Mod. Phys.* **88** (2016) 035006.
[26] Shang, Y., On the number of spanning trees, the Laplacian eigenvalues, and the Laplacian Estrada index of subdivided-line graphs, *Open Math.* **14** (2016) 641–648.
[27] Newman, M .E. J., Strogatz, S. H. and Watts, D. J., Random graphs with arbitrary degree distributions and their applications, *Phys. Rev. E* **64** (2001) 026118.
[28] Mézard, M. and Parisi, G., The Bethe lattice spin glass revisited, *Eur. Phys. J. B* **20** (2001) 217–233.
[29] Weigt, M. and Hartmann, A. K., *Phase Transitions in Combinatorial Optimization*

Problems, Wiley-VCH, Weinheim, 2005.
[30] Boldi, P., Rosa, M., Santini, M. and Vigna, S., Layered label propagation: A multiresolution coordinate-free ordering for compressing social networks, *International World Wide Web Conference*, ACM, Hyderabad, India, pp.587–596, 2011.
[31] Palka, Z., Extreme degrees in random graphs, *J. Graph Theory* **11** (1987) 121–134.
[32] Broido, A. D. and Clauset, A., Scale-free networks are rare, *Nat. Commun.* **10** (2019) 1017.
[33] Amunts, K., Lepage, C., Borgeat, L., Mohlberg, H., Dickscheid, T., Rousseau, M., Bludau, S., Bazin, P.-L., Lewis, L. B., Oros-Peusquens, A.-M., Shah, N. J., Lippert, T., Zilles, K. and Evans, A. C., BigBrain: An Ultrahigh-Resolution 3D Human Brain Model, *Science* **340** (2013) 1472–1475.