L-hop percolation on networks with arbitrary degree distributions and its applications

Yilun Shang,[1] Weiliang Luo,[2] and Shouhuai Xu[2]
[1]*Institute for Cyber Security, University of Texas at San Antonio, Texas 78249, USA*
[2]*Department of Computer Science, University of Texas at San Antonio, Texas 78249, USA*
(Received 22 March 2011; revised manuscript received 4 July 2011; published xxxxx)

Site percolation has been used to help understand analytically the robustness of complex networks in the presence of random node deletion (or failure). In this paper we move a further step beyond random node deletion by considering that a node can be deleted because it is chosen or because it is within some L-hop distance of a chosen node. Using the generating functions approach, we present analytic results on the percolation threshold as well as the mean size, and size distribution, of nongiant components of complex networks under such operations. The introduction of parameter L is both conceptually interesting because it accommodates a sort of *nonindependent* node deletion, which is often difficult to tackle analytically, and practically interesting because it offers useful insights for cybersecurity (such as botnet defense).

DOI: 10.1103/PhysRevE.00.001100 PACS number(s): 02.50.−r, 64.60.−i, 89.75.Hc

I. INTRODUCTION

Understanding the robustness (or connectivity) of complex networks in the presence of node deletion (or failure) is an important problem that has attracted much attention. Existing analytic studies mainly focused on the case of randomly deleting nodes in random graphs with a given arbitrary degree distribution (the so-called configuration model [1–5]). By treating node deletion as site percolation on large networks (i.e., in the limit of network size), Newman *et al.* [3] introduced a novel generating functions approach to tackling the problem. This approach has been used in various settings (e.g., Refs. [6–8]).

In this paper, we move a further step beyond random node deletion by considering the following node deletion operation: a node is deleted either because it is chosen or because it is within a chosen node's L-hop distance in terms of shortest path between them in unweighted graphs, where $L \geqslant 0$. Figure 1 illustrates an example scenario with $L = 1$, where, if v is chosen to be deleted, then both v and its one-hop (i.e., direct) neighbors u_1,u_2,u_3,u_4 are deleted. We call the new model "L-hop percolation" because the random node deletion extensively studied in the literature [3–8] can be seen as the special case with $L = 0$. The introduction of the new parameter L is of both conceptual and practical value, as we discuss below.

From a conceptual perspective, the introduction of L allows us to model and characterize a class of *nonindependent* node deletion operations. Previous analytical studies (including Refs. [3–8]) in this context focused on *independent* node deletions. Nonindependent node deletions are often challenging to treat analytically, but we manage to present some analytical results in the specific nonindependence setting of this paper.

From a practical perspective, the nonindependent node deletion has real-life applications. On the one hand, an adversary could certainly destroy both randomly chosen nodes and their neighbors within L-hop distance. On the other hand, analyzing such nonindependent node deletions can lead to useful insights for defending cyberspace. To see this, consider an example of botnet defense. A botnet is a network of computers (called bots or zombies) that are compromised by a computer malware/virus and under the control of a bot-master (the human attacker). Peer-to-peer (P2P) botnets [9] are arguably the most powerful cyberattacks that are very difficult to defend against. This is because a bot only "knows" (i.e., can directly communicate with) a few others, which suffices their needs of forwarding the bot-master's command-and-control messages. In real life, it is very difficult to obtain the precise structures of P2P botnets because of various technical issues and ethics/privacy concerns [10,11]. This means that the defender cannot expect to identify all the bots belonging to a botnet and then eliminate them as a whole. Instead, it is more realistic that the defender, first, detects some bots and then traces the bots that communicated with the detected ones (we call this method "detect and then trace"). However, the defender's capability in tracing bots are limited by various reasons. For example, botmasters have abused cryptographic techniques for making P2P botnets—such as Conficker, Nugache, and Storm and its successor, Waledac—more stealthy [9,12–15]. Moreover, the Waledac botnets actually do not spread (propagate) using a scanning technique that is relatively easy to defend against but rather spread (propagate) via social-engineering methods that are much more difficult to deal with [16]. Because P2P botnets are naturally complex networks, the defender's limited tracing capability in combating P2P botnets naturally motivates the concept of L-hop percolation. Note that $L = 0$ corresponds to the fact that the defender can only directly detect (and thus delete) some bots, and $L \geqslant 0$ corresponds to the fact that the defender not only can directly detect (and thus delete) some bots but also can trace to (and thus delete) the bots within their L-hop neighborhood.

The present paper aims to understand and characterize the robustness of complex networks with $L > 0$ when compared to the case with $L = 0$. Specifically, we consider L-hop site percolation on random graphs with arbitrary node degree distributions (in the configuration model). Using the generating functions approach [3,6], we present analytical results on the percolation threshold, and mean size as well as size distribution of the nongiant components. The analytical results are confirmed by our simulation study on power-law random graphs and Erdos-Renyi random graphs.

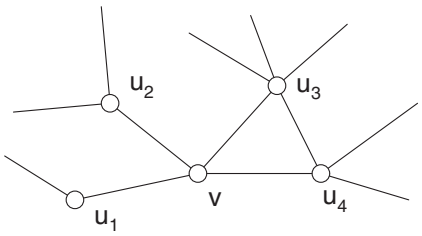


FIG. 1. An example scenario: when $L = 0$, a randomly chosen node v is deleted (this is the case extensively studied in the literature); when $L = 1$, a randomly chosen node v and its one-hop neighbors u_1,u_2,u_3,u_4 are deleted. As defined in the text, $\mathcal{N}_{\leqslant 0}(v) = \{v\}$ and $\mathcal{N}_{\leqslant 1}(v) = \{v,u_1,u_2,u_3,u_4\}$.

II. SYSTEM MODEL AND ANALYTIC RESULTS

Let $\mathsf{G} = (\mathsf{V},\mathsf{E})$ be a complex network (or graph), where $v \in \mathsf{V}$ is the set of nodes (vertices) and E is the set of edges (we focus on undirected unweighted graphs). Let $\mathsf{distance}(u,v)$ be the length of the shortest path between $u \in \mathsf{V}$ and $v \in \mathsf{V}$ in G, where $\mathsf{distance}(v,v) = 0$. Define the ℓ-hop neighborhood of node v as $\mathcal{N}_{\leqslant \ell}(v) = \{u : u \in \mathsf{V}, \mathsf{distance}(u,v) \leqslant \ell\}$, namely the set of nodes that are no more than ℓ-hop away from v. The following parameters are used in our model:

(1) L: This is upper bound on the number of hops the deletion operation can reach out to when starting at a randomly chosen node. Note that L indicates, in a sense, the degree of nonindependence of the nodes that are deleted (intuitively, the larger the L, the stronger the nonindependence) and can represent (for instance) the defender's capability in tracing bots in the above example of botnet defense.

(2) $(r_0,r_1,\ldots,r_L)$ with $\sum_{\ell=0}^{L} r_\ell = 1$: For a node v and its L-hop neighborhood $\mathcal{N}_{\leqslant L}(v)$, the simplest scenarios is to delete all the nodes belonging to $\mathcal{N}_{\leqslant \ell}(v)$, which is a special case we mentioned above. To make our model more generally applicable while reflecting the fact mentioned above—the farer from v the neighbors, the less likely the defender can trace to them—we need a representation of detection probability that decreases as the distance from a chosen node grows. For this, we use vector $(r_0,r_1,\ldots,r_L)$ to represent a class of deletion "strategies" such that a node u, which is $\mathsf{distance}(u,v)$-hop away from a chosen node v, will be deleted with probability $\sum_{\ell=\mathsf{distance}(u,v)}^{L} r_\ell$. Because the chosen node v will be deleted with probability $\sum_{\ell=0}^{L} r_\ell$, we naturally require $\sum_{\ell=0}^{L} r_\ell = 1$, meaning that v will always be deleted. This explains why the extensively studied case with $L = 0$ is a special case of the model with $L \geqslant 0$ in this paper. In order to see that the representation corresponds to a class of deletion operations, we note that the scenario of $L = 0$ shown in Fig. 1 corresponds to strategy $(r_0) = (1)$, and the scenario of $L = 1$ corresponds to strategy $(r_0,r_1) = (0,1)$. Continuing on the example of $L = 1$ but with a different strategy $(0 < r_0 < 1, 0 < r_1 < 1)$ where $r_0 + r_1 = 1$, we observe that node v will be deleted with probability $r_0 + r_1 = 1$ but every node belonging to $\mathcal{N}_{\leqslant 1}(v) - \mathcal{N}_{\leqslant 0}(v) = \{u_1,u_2,u_3,u_4\}$ will be deleted with probability r_1.

(3) p_k: This is the probability that a randomly chosen node has degree k.

(4) q, q_k, q_c: q is the probability an appropriately chosen node is *occupied* (i.e., not deleted), and $1 - q$ is the probability that node is *unoccupied* (i.e., deleted). In general $0 \leqslant q \leqslant 1$ because, for example, the curing of a bot may not be perfect [17]. Correspondingly, q_k is the probability that a node of degree k is occupied, and q_c is the percolation threshold we want to derive, above which a giant component emerges.

Suppose $|x| \leqslant 1$. Following the generating functions approach [3], $G_0(x) = \sum_{k=0}^{\infty} p_k x^k$ and $G_1(x) = G_0'(x)/G_0'(1)$ are the generating functions of nodes' degree distribution and nodes' excess-degree distribution, respectively. For any node $v \in \mathsf{V}$, the generating function for the size distribution of $\mathcal{N}_{\leqslant \ell}(v) - \mathcal{N}_{\leqslant \ell-1}(v)$ is $G^{(\ell)}(x) := G_0[G_1(\cdots G_1(x)\cdots)]$, with $\ell - 1$ iterations of the function G_1 acting on itself. For a given L, for each $0 \leqslant \ell \leqslant L$ we can define a "base" strategy $\mathsf{baseStrategy}_\ell = (\mathsf{r}_0,\ldots,\mathsf{r}_L)$, where $\mathsf{r}_j = 1$ if $j = \ell$ and $\mathsf{r}_j = 0$ otherwise. (This is for deriving some intermediate formulas that will be used to derive the final result.) Let $q_k^{(\ell)}$ be the probability that a node of degree k is occupied (i.e., not deleted) with respect to $\mathsf{baseStrategy}_\ell$. Note that the component structure in graph G is essentially treelike because the probability that a component contains a closed loop goes as N^{-1} and is therefore negligible in the limit of large N. Then, for a node v of degree k, we have $q_k^{(0)} = q$, $q_k^{(1)} = q^{k+1}$, and for $\ell \geqslant 2$,

$$q_k^{(\ell)} = \sum_{s_1=0}^{\infty} \cdots \sum_{s_{\ell-1}=0}^{\infty} \left(q^{k+1+\sum_{j=1}^{\ell-1} s_j} \prod_{j=1}^{\ell-1} \frac{d^{s_j} G^{(j+1)}(x)}{s_j!\, dx^{s_j}}\bigg|_{x=0} \right), \quad (1)$$

where s_j is the number of v's distance-$(j+1)$ neighbors [i.e., s_j is the size of the set $\{u : \mathsf{distance}(u,v) = j+1\}$], $\frac{d^{s_j} G^{(j+1)}(x)}{s_j! dx^{s_j}}|_{x=0}$ is the probability that v has exactly s_j distance-$(j+1)$ neighbors (based on the property of derivatives of generating functions for degree distributions [3,18]), $\prod_{j=1}^{\ell-1} \frac{d^{s_j} G^{(j+1)}(x)}{s_j! dx^{s_j}}|_{x=0}$ is the probability that v has respectively s_j distance-$(j+1)$ neighbors for $j = 1,\ldots,\ell-1$, and $q^{k+1+\sum_{j=1}^{\ell-1} s_j} \prod_{j=1}^{\ell-1} \frac{d^{s_j} G^{(j+1)}(x)}{s_j!\, dx^{s_j}}|_{x=0}$ is the probability that v is occupied (i.e., not deleted) with respect to $\mathsf{baseStrategy}_\ell$ for a specific ℓ.

Given $q_k^{(\ell)}$ for $\ell = 0,1,\ldots$ and strategy $(r_1,\ldots,r_L)$, we obtain

$$q_k = \sum_{\ell=0}^{L} q_k^{(\ell)} r_\ell. \quad (2)$$

The probability that any node of degree k is occupied is given by $p_k q_k$, and

$$F_0(x) = \sum_{k=0}^{\infty} p_k q_k x^k \quad (3)$$

is the probability generating function for this distribution [18]. Since $F_0(1)$ is the probability that a randomly chosen node is occupied, it can be viewed as the portion of occupied nodes.

As in Refs. [3,6], if we start at a randomly chosen node and follow each of its edges to reach its neighbors, the distribution of the other edges of each node arrived and occupied is generated by

$$F_1(x) = \frac{\sum_{k=0}^{\infty} k p_k q_k x^{k-1}}{\sum_{k=0}^{\infty} k p_k} = \frac{F_0'(x)}{z}, \quad (4)$$

where $z = \langle k \rangle = G_0'(1)$ is the average node degree.

Let $H_1(x)$ be the generating function for the distribution of the sizes of percolation components with occupied nodes that are reached by choosing a random edge and following it to one of its ends. With the same sort of reasoning as in Ref. [7], we derive that $H_1(x)$ satisfies the self-consistency condition

$$H_1(x) = 1 - F_1(1) + xF_1[H_1(x)]. \tag{5}$$

Similarly, the distribution for the size of the component to which a randomly chosen node belongs is generated by

$$H_0(x) = 1 - F_0(1) + xF_0[H_1(x)], \tag{6}$$

which, in conjunction with Eqs. (1)–(5), can be used to determine some quantities of interest such as the L-hop percolation threshold, the mean size, and size distribution of nongiant components. In what follows we present the details.

A. L-hop percolation threshold

From Eqs. (5) and (6), we obtain that in the absence of giant components, the mean size of components to which a randomly chosen node belongs is given by

$$\begin{aligned}\langle s\rangle &= H_0'(1) = F_0(1) + F_0'(1)H_1'(1)\\ &= F_0(1) + \frac{F_0'(1)^2}{z - F_0''(1)}.\end{aligned} \tag{7}$$

The above expression diverges at $z = F_0''(1)$, which corresponds to the percolation threshold q_c (i.e., the critical occupation probability) where a giant component emerges. Hence, q_c is determined by

$$\sum_{k=0}^{\infty} kp_k = \sum_{k=0}^{\infty} k(k-1)p_k q_k, \tag{8}$$

where q_k is given by Eq. (1). In what follows we instantiate the above general q_c in the special cases of regular graphs and of the classical Erdos-Renyi random graphs.

In the case of d-regular graphs G on n nodes, where d is a positive integer, we have the degree distribution $p_d = 1$ and $p_k = 0$ for $k \neq d$. From Eqs. (1) and (2) we get

$$q_d = \sum_{\ell=0}^{L} q_d^{(\ell)} r_\ell, \tag{9}$$

where for $\ell = 0, \ldots, L$,

$$q_d^{(\ell)} = \sum_{s_1=0}^{\infty} \cdots \sum_{s_{\ell-1}=0}^{\infty} \left(q^{d+1+\sum_{j=1}^{\ell-1} s_j} \prod_{j=1}^{\ell-1} \frac{d^{s_j} G^{(j+1)}(x)}{s_j!\, dx^{s_j}}\bigg|_{x=0} \right). \tag{10}$$

By utilizing Eq. (8), we see that the percolation threshold q_c can be calculated from

$$1 = (d-1)q_d. \tag{11}$$

In the case of the classical Erdos-Renyi random graphs $G(n,\lambda/n)$ for some $\lambda > 0$ [19], the average degree for each vertex is λ and the degree distribution is given by $p_k = e^{-\lambda}\lambda^k/k!$ for $k \geqslant 0$. From Eq. (8), we see that the percolation threshold q_c is determined by

$$\sum_{k=2}^{\infty} \frac{\lambda^k}{(k-2)!} \left(\frac{1}{k-1} - q_k \right) = 0. \tag{12}$$

where q_k is given by (2) and (1).

B. Mean size of nongiant components

Because $H_0(x)$ generates the size distribution of nongiant components, $H_0(1)$ takes the value of $1 - S$, where S is the fraction of nodes belonging to some giant component (equivalently, S is the fraction of giant components). Therefore,

$$S = 1 - H_0(1) = F_0(1) - F_0(u), \tag{13}$$

where $u = H_1(1)$ is the smallest non-negative solution of

$$u = 1 - F_1(1) + F_1(u). \tag{14}$$

In general, the mean size of the component, excluding the infinite giant component, to which a randomly chosen vertex belongs can be expressed as

$$\begin{aligned}\langle s\rangle &= \frac{H_0'(1)}{H_0(1)} = \frac{F_0[H_1(1)] + F_0'[H_1(1)]\left\{\frac{F_1[H_1(1)]}{1-F_1'[H_1(1)]}\right\}}{H_0(1)}\\ &= \frac{F_0(u)[z - F_0''(1)] + F_0'(u)^2}{(1-S)[z - F_0''(1)]},\end{aligned} \tag{15}$$

which is equivalent to Eq. (7) when there is no giant component (i.e., $S = 0$, $u = 1$).

C. Size distribution of nongiant components

Let π_s be the probability of a randomly chosen node belonging to a component of size s. We can then rewrite the generating function $H_0(x)$ as

$$H_0(x) = \pi_0 + \sum_{s=1}^{\infty} \pi_s x^s. \tag{16}$$

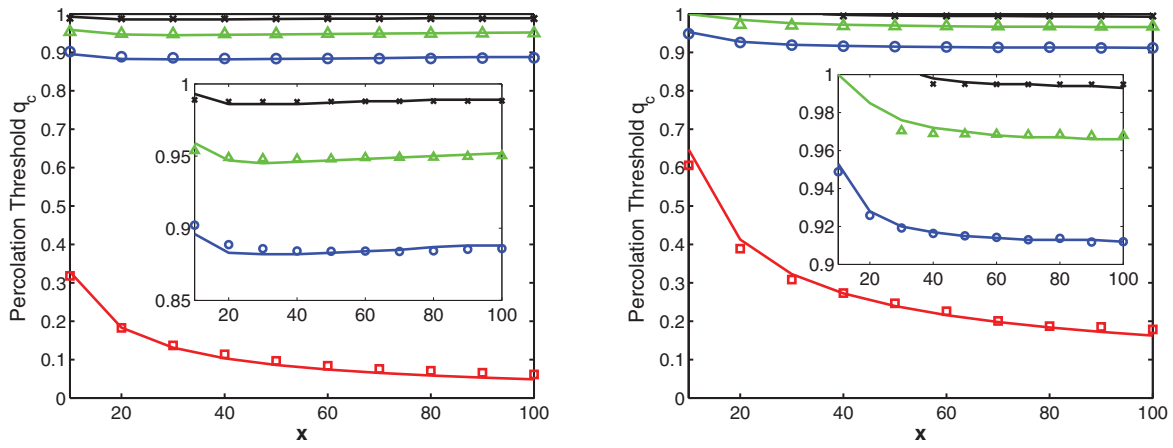
It is straightforward to see that $\pi_0 = 1 - F_0(1)$ and $\pi_1 = H_0(1) - \pi_0 - \sum_{s=2}^{\infty} \pi_s$, where $H_0(1)$ can be determined by Eqs. (13) and (14). For $s \geqslant 2$, we have from Eqs. (16) and (6)

$$\begin{aligned}\pi_s &= \frac{1}{(s-1)!} \left\{ \frac{d^{s-1}}{dx^{s-1}} \left[\frac{H_0(x) - \pi_0}{x} \right] \right\} \bigg|_{x=0}\\ &= \frac{1}{(s-1)!} \frac{d^{s-1}}{dx^{s-1}} \{F_0[H_1(x)]\}|_{x=0}\\ &= \frac{1}{(s-1)!} \frac{d^{s-2}}{dx^{s-2}} [F_0'(H_1(x))H_1'(x)]|_{x=0}.\end{aligned} \tag{17}$$

By using the Cauchy formula for the higher-order derivative of a function [20],

$$\frac{d^n f}{dz^n}\bigg|_{z=z_0} = \frac{n!}{2\pi i} \oint \frac{f(z)}{(z-z_0)^{n+1}} dz, \tag{18}$$

where the integral is around a contour that encloses z_0 in the complex plane but encloses no poles in $f(z)$. Equation (17)



(a) $t = 2$ (b) $t = 2.5$

FIG. 2. (Color online) Percolation threshold q_c in the case of power-law graphs. Simulation results are plotted with respect to $L = 0$ with strategy $(r_0) = (1)$ (squares), $L = 1$ with strategy $(r_0,r_1) = (0,1)$ (circles), $L = 2$ with strategy $(r_0,r_1,r_2) = (0,0,1)$ (triangles), and $L = 3$ with strategy $(r_0,r_1,r_2,r_3) = (0,0,0,1)$ (crosses). The insets show a magnified view of the curves for $L = 1$, $L = 2$, and $L = 3$. In the simulation we set the threshold for giant component as 1000 nodes. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eq. (8).

then becomes

$$\pi_s = \frac{1}{2\pi i(s-1)} \oint \frac{F_0'[H_1(x)]}{x^{s-1}} \frac{dH_1(x)}{dx} dx \qquad (19)$$

$$= \frac{z}{2\pi i(s-1)} \oint \frac{F_1(H_1)}{x^{s-1}} dH_1, \qquad (20)$$

where the integral in Eq. (19) is around an infinitesimal contour around the origin in the complex plane, and the integral in Eq. (20) is an infinitesimal loop around $1 - F_1(1)$ since $H_1(x) \to 1 - F_1(1)$ as $x \to 0$.

By treating x as a function of H_1, we plug Eq. (5) into Eq. (20) to obtain

$$\pi_s = \frac{z}{2\pi i(s-1)} \oint \frac{[F_1(H_1)]^s}{[H_1 - 1 + F_1(1)]^{s-1}} dH_1. \qquad (21)$$

By applying the Cauchy formula to Eq. (21), we obtain the desired result

$$\pi_s = \frac{z}{(s-1)!} \left\{ \frac{d^{s-2}}{dx^{s-2}} [F_1(x)^s] \right\} \Big|_{x=1-F_1(1)}. \qquad (22)$$

III. SIMULATION RESULTS

A. Simulation-based confirmation of the percolation thresholds

In order to verify the theoretical results, we first conducted simulations on random graphs at the order 1 000 000 nodes with node degrees distributed according to the truncated power law $p_0 = 0$ and

$$p_k = Ck^{-t} e^{-k/x}, \qquad k \geqslant 1, \qquad (23)$$

where C, t, and x are constants. The reason for choosing this distribution is twofold. On one hand, this distribution is seen in various real-life networks including the Internet [21], collaboration networks of movie actors and scientists [22,23]. On the other hand, the exponential cutoff makes the distribution normalizable for all t so the generating functions and their derivatives are finite. In our simulation study we set $0 \leqslant L \leqslant 3$ and, for simplicity, strategy $(r_0, \ldots, r_L) = (0, \ldots, 0, 1)$. We consider two scenarios of $t = 2$ and $t = 2.5$.

Figure 2 plots the percolation threshold q_c obtained from our simulations, along with the exact solutions obtained from Eq. (8). We make the following observations. First, the simulation results agree with their analytical counterparts. Second, there is a huge gap between the percolation thresholds

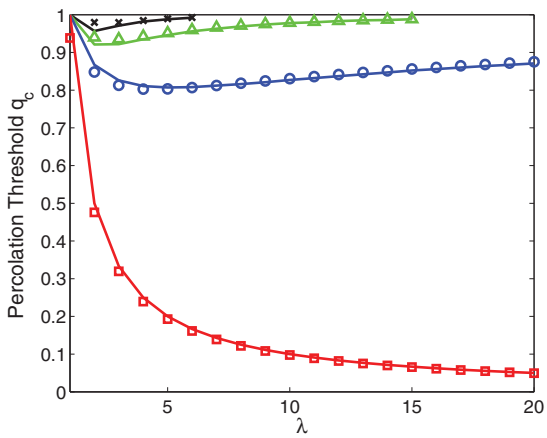


FIG. 3. (Color online) Percolation threshold in the case of Erdos-Renyi graphs. Simulation results are plotted with respect to $L = 0$ with strategy $(r_0) = (1)$ (squares), $L = 1$ with strategy $(r_0,r_1) = (0,1)$ (circles), $L = 2$ with strategy $(r_0,r_1,r_2) = (0,0,1)$ (triangles), and $L = 3$ with strategy $(r_0,r_1,r_2,r_3) = (0,0,0,1)$ (crosses). In the simulation we set the threshold for giant component as 1000 nodes. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eq. (12).

with respect to $L = 0$ and $L = 1$. For example, consider the case of $t = 2$ with $x = 50$. When $L = 0$, giant component disappears only after randomly deleting 90% nodes; whereas, when $L = 1$, giant component disappears after deleting 12% nodes, including both the randomly chosen nodes and their direct neighbors. This means, in the botnet defense example, that "detect and then trace" could be much more effective.

Third, there is no big difference between the thresholds corresponding to $L = 1,2,3$, respectively. This phenomenon can be interpreted as follows: In networks with a power-law degree distribution, most nodes have small degrees and only a small number of nodes (called hubs) have large degrees. Therefore, a randomly chosen node probably has a neighbor that is a hub, meaning that when $L = 1$ with strategy $(\mathsf{r}_0,\mathsf{r}_1) = (0,1)$, a hub could be deleted even after choosing a small number of nodes. This explains the abrupt change in the percolation threshold in the case of $L = 1$ with strategy $(\mathsf{r}_0,\mathsf{r}_1) = (0,1)$ when compared with the percolation threshold in the case of $L = 0$ with $\mathsf{r}_0 = 1$. The fact that $L = 1$ already allows the deletion of some hub nodes further explains why increasing L further may not significantly increase the percolation threshold. This means that tracing one-hop neighbors is almost as powerful as tracing multihop neighbors. We mention that $L = 1$ with

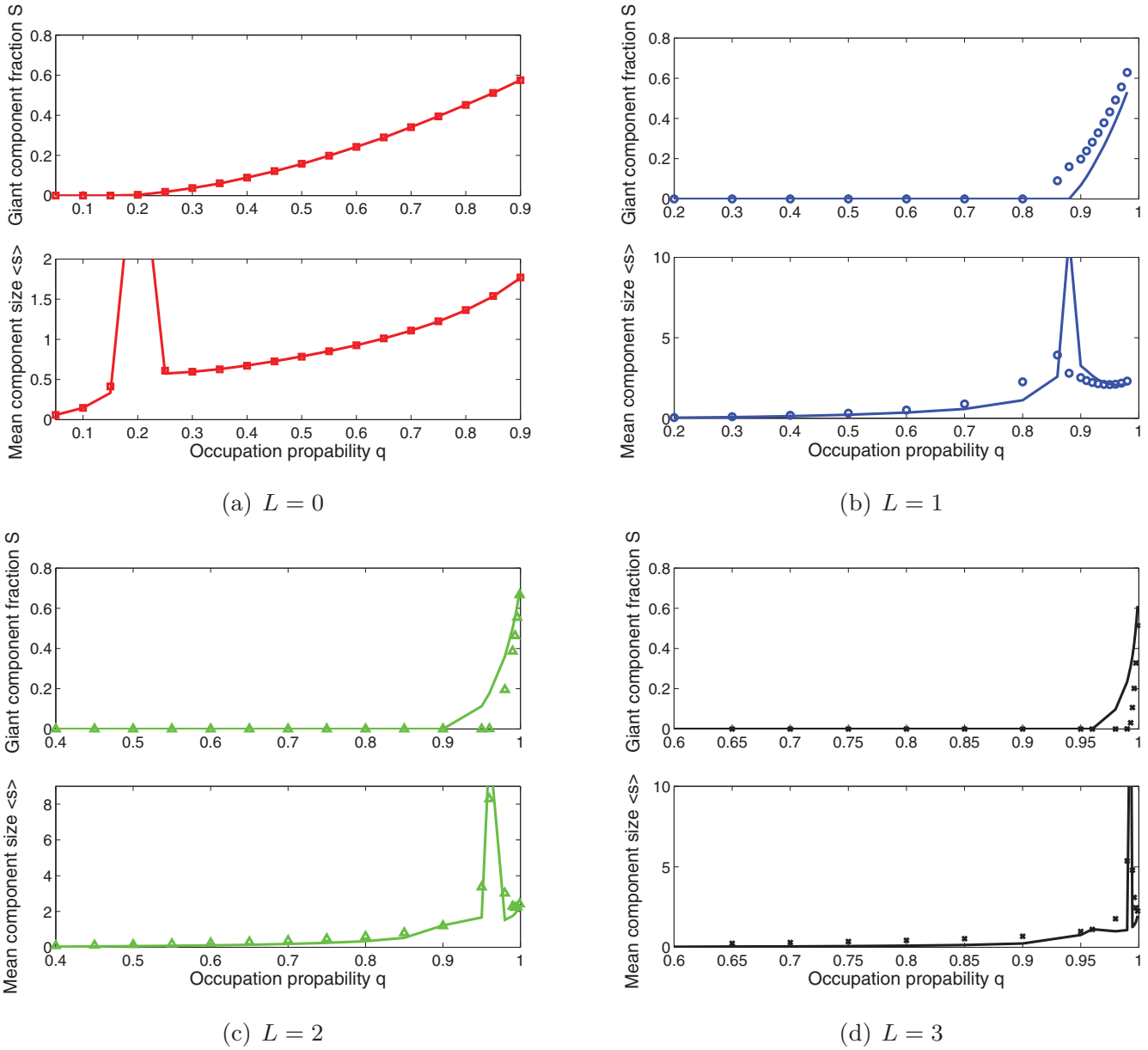


FIG. 4. (Color online) Fraction of giant components S and mean size of nongiant components $\langle s \rangle$ in power-law random graphs with $t = 2$ and $x = 20$. Simulation results are plotted with respect to (a) $L = 0$ with strategy $(\mathsf{r}_0) = (1)$ (squares), (b) $L = 1$ with strategy $(\mathsf{r}_0,\mathsf{r}_1) = (0,1)$ (circles), (c) $L = 2$ with strategy $(\mathsf{r}_0,\mathsf{r}_1,\mathsf{r}_2) = (0,0,1)$ (triangles), and (d) $L = 3$ with strategy $(\mathsf{r}_0,\mathsf{r}_1,\mathsf{r}_2,\mathsf{r}_3) = (0,0,0,1)$ (crosses). In the simulation, we set the threshold for giant component as 1000 nodes. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eqs. (13), (14), and (15).

strategy $(r_0, r_1) = (0,1)$ is somewhat related to the method known as "acquaintance immunization" [24], which consists of choosing a random node and deleting a random one-hop neighbor of the chosen node (the chosen node itself is not deleted). It was shown in Ref. [24] that the effectiveness of this method (especially on power-law networks) comes from the fact that with high probability the randomly chosen neighbor of a randomly chosen node is a hub, and therefore the hubs are knocked out after a small number of such operations. In this paper, the same effect causes the dramatic increase in q_c when changing $L = 0$ to $L = 1$.

We then conducted simulations on the classic Erdos-Renyi graphs $G(n,p)$ with $n = 1\,000\,000$ nodes and link probability $p = \lambda/n$. Figure 3 plots the percolation threshold q_c obtained from our simulations, along with the exact solutions obtained from Eq. (12). Similar qualitative behavior can be observed as in the case of power-law graphs, with two noticeable differences: the difference between the thresholds corresponding to $L = 0$ and $L = 1,2,3$ trails off, while the differences between $L = 1,2,3$ become more apparent. These phenomena can be explained by that Erdos-Renyi random graphs have more homogeneous degree distributions than power-law graphs.

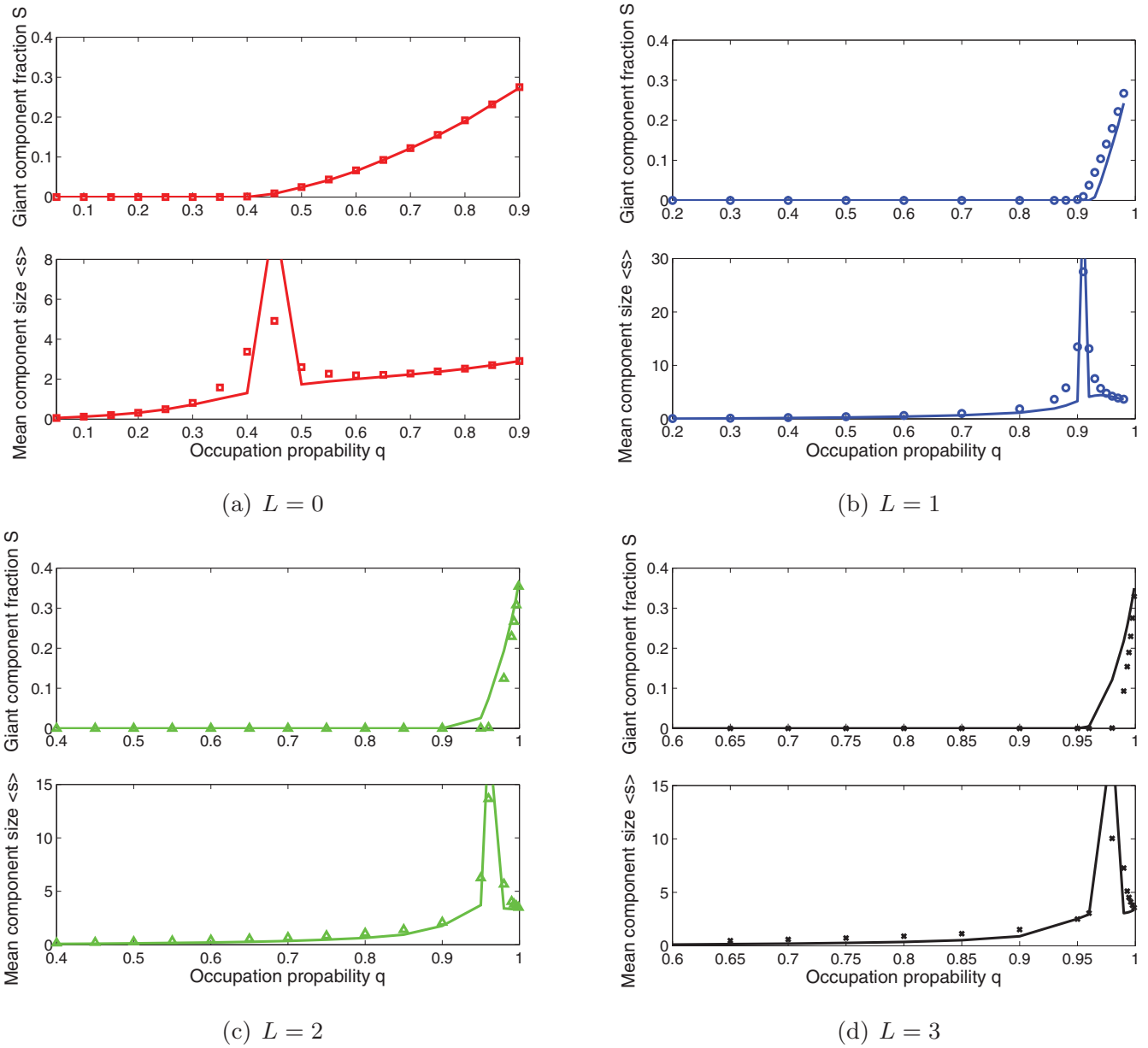


FIG. 5. (Color online) Fraction of giant components S and mean size of nongiant components $\langle s \rangle$ in power-law random graphs with $t = 2.5$ and $x = 20$. Simulation results are plotted with respect to (a) $L = 0$ with strategy $(r_0) = (1)$ (squares), (b) $L = 1$ with strategy $(r_0, r_1) = (0,1)$ (circles), (c) $L = 2$ with strategy $(r_0, r_1, r_2) = (0,0,1)$ (triangles), and (d) $L = 3$ with strategy $(r_0, r_1, r_2, r_3) = (0,0,0,1)$ (crosses). In the simulation, we set the threshold for giant component as 1000 nodes. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eqs. (13), (14), and (15).

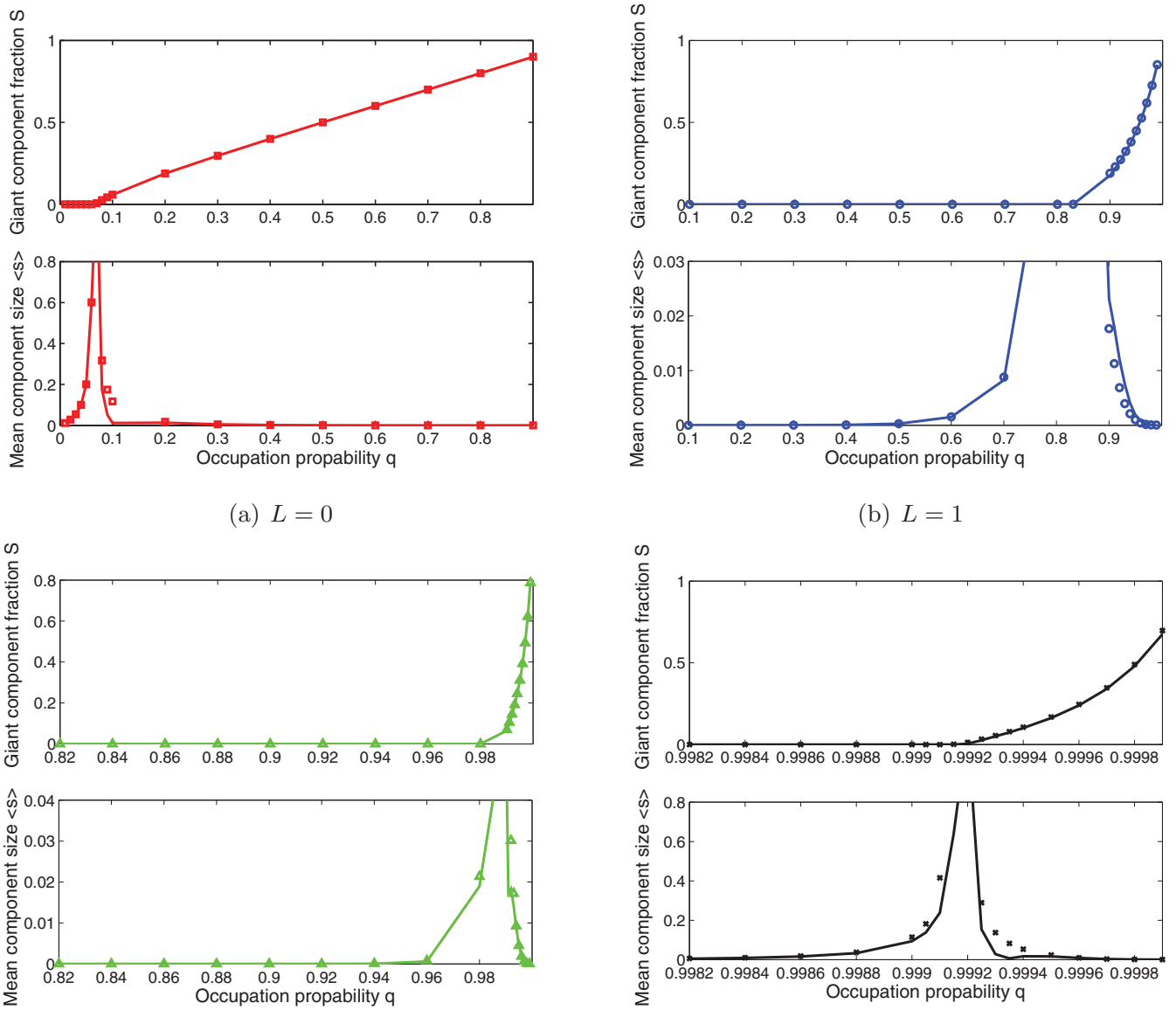
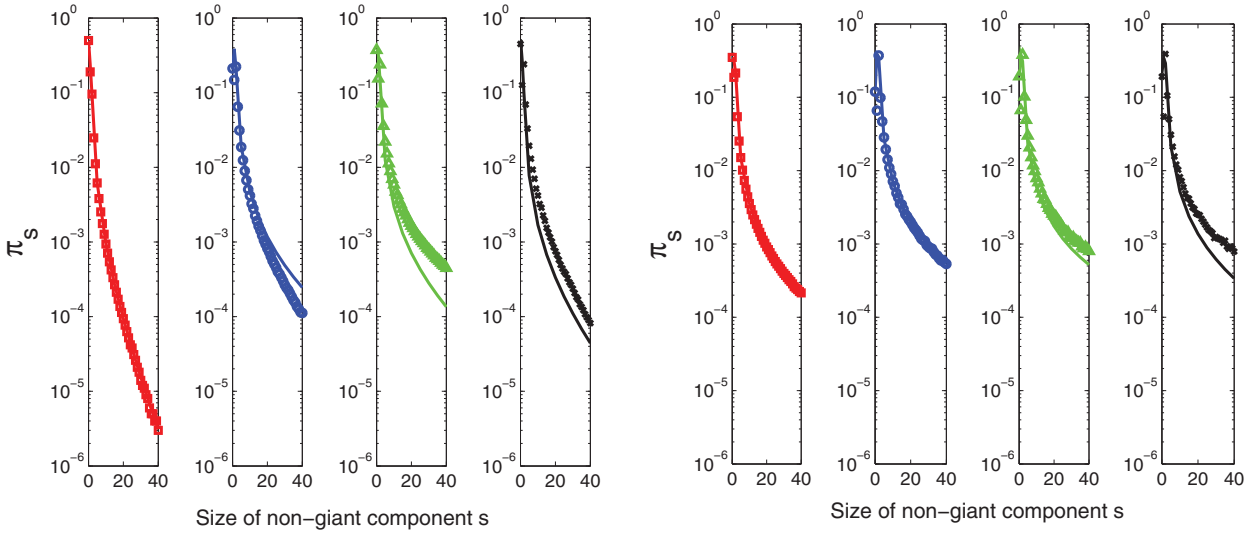


FIG. 6. (Color online) Fraction of giant components S and mean size of nongiant components $\langle s \rangle$ in Erdos-Renyi random graphs with $\lambda = 15$. Simulation results are plotted with respect to (a) $L = 0$ with strategy $(r_0) = (1)$ (squares), (b) $L = 1$ with strategy $(r_0, r_1) = (0,1)$ (circles), (c) $L = 2$ with strategy $(r_0, r_1, r_2) = (0,0,1)$ (triangles), and (d) $L = 3$ with strategy $(r_0, r_1, r_2, r_3) = (0,0,0,1)$ (crosses). In the simulation we set the threshold for giant component as 1000 nodes. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eqs. (13), (14), and (15).

B. Simulation-based confirmation of the fraction S of giant components and the mean size $\langle s \rangle$ of the nongiant components

For this purpose, we conducted simulations on the same power-law random graphs with distribution (23) and the same Erdos-Renyi random graphs $G(n, \lambda/n)$ with $n = 1\,000\,000$. Figure 4 and Fig. 5 plot the cases of power-law graphs with $t = 2, x = 20$ and $t = 2.5, x = 20$, respectively. Figure 6 plots the case of Erdos-Renyi random graphs with $\lambda = 15$. From them we draw the following observations.

First, we observe that when the occupation probability $q < q_c$, the fraction of giant components is $S = 0$. Then, S increases with q when $q \geqslant q_c$ for all four strategies $(r_0, \ldots, r_L) = (0, \ldots, 0, 1)$ with respect to $L = 0, 1, 2, 3$. More interestingly, from another perspective, we observe that as the number of chosen nodes increases, the fraction of giant components decreases abruptly in the cases of $L > 1$ but decreases smoothly in the case of $L = 0$. This illustrates the power of deletion of even one-hop neighbors. Moreover, the giant component disappears faster in the case of power-law graphs than in the case of Erdos-Renyi random graphs. For example, in the case of $L = 1$, the giant component disappears after deleting about 10% (20%) of nodes and their one-hop neighbors in the case power-law graphs with $t = 2.5$ as shown in Fig. 5(b) [in the case of Erdos-Renyi random graphs as shown in Fig. 6(b)]. This can be explained, as mentioned above,



(a) Power-law graphs with $t = 2, x = 20$ (b) Power-law graphs with $t = 2.5, x = 20$

FIG. 7. (Color online) Size distribution for nongiant components π_s in power-law random graphs. Simulation results are plotted with respect to, from left to right, $L = 0$ with strategy $(r_0) = (1)$ (squares), $L = 1$ with strategy $(r_0, r_1) = (0,1)$ (circles), $L = 2$ with strategy $(r_0, r_1, r_2) = (0,0,1)$ (triangles), and $L = 3$ with strategy $(r_0, r_1, r_2, r_3) = (0,0,0,1)$ (crosses). In the simulation we set in (a) the occupation probabilities $q = 0.5, 0.92, 0.95, 0.98$ to keep the fraction of giant components $S \approx 0.11$ in all four cases; we set in (b) $q = 0.65, 0.95, 0.96, 0.98$ to keep the fraction of giant components $S \approx 0.09$ in all four cases. The simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eq. (22).

by the fact that in the case of power-law graphs, it is relatively easier to hit the hubs because they are the neighbors of many nodes.

Second, when q approaches the critical point q_c, the mean size of nongiant components $\langle s \rangle$ increases abruptly. This agrees with our theory. In fact, when $q = q_c$ we obtain $z = F_0''(1)$ in Eq. (21). By using Eq. (15), we obtain $\lim_{q \to q_c} \langle s \rangle = \infty$. The percolation thresholds q_c shown in Fig. 4, Fig. 5, and Fig. 6 agree with those in Fig. 2(a), Fig. 2(b), and Fig. 3. By comparing Fig. 4 and Fig. 5 with Fig. 6, we make two observations. (i) In power-law graphs, typical nongiant components have multiple nodes. In Erdos-Renyi random graphs, however, most nongiant components are isolated nodes because $\langle s \rangle$ is always less than 1 (when unoccupied nodes are treated as components of size zero). (ii) The analytical and numerical results match better in the case of Erdos-Renyi random graphs than in the case of power-law graphs, due to the inhomogeneity of power-law degree distribution.

C. Simulation-based confirmation of the size distribution π_s of nongiant components

To illustrate this, we conducted simulations on the same power-law random graphs with distribution (23) and the same Erdos-Renyi random graphs $G(n, \lambda/n)$ with $n = 1\,000\,000$. We plot the log-linear diagrams in Fig. 7 for power-law graphs and in Fig. 8 for Erdos-Renyi random graphs.

In Fig. 7(a) and Fig. 7(b), we keep the respective fractions of giant components at $S \approx 0.11$ and $S \approx 0.09$ in all strategies $L = 0,1,2,3$ (this can be achieved by choosing different occupation probabilities q). In Fig. 8, we keep the fraction of giant components at $S \approx 0.07$ in all strategies $L = 0,1,2,3$ (also by choosing different occupation probabilities q).

By comparing Fig. 7 and Fig. 8, we observe the following discrepancy between power-law graphs and Erdos-Renyi graphs. For power-law graphs with the same fraction of

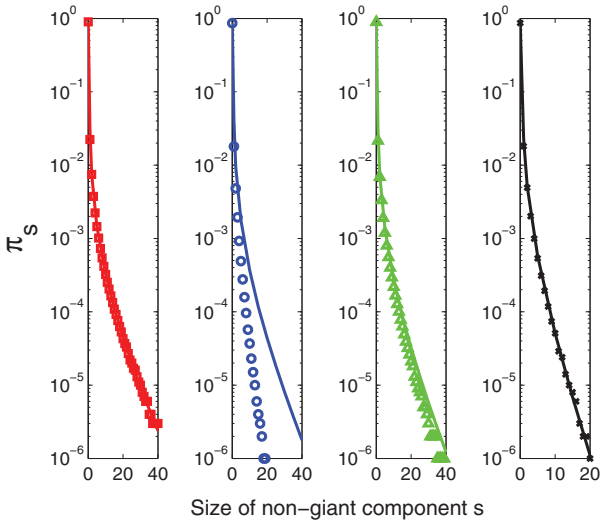


FIG. 8. (Color online) Size distribution for nongiant components π_s in Erdos-Renyi random graphs with $\lambda = 15$. Simulation results are plotted with respect to, from left to right, $L = 0$ with strategy $r_0 = 1$ (squares), $L = 1$ with strategy $(r_0, r_1) = (0,1)$ (circles), $L = 2$ with strategy $(r_0, r_1, r_2) = (0,0,1)$ (triangles), and $L = 3$ with strategy $(r_0, r_1, r_2, r_3) = (0,0,0,1)$ (crosses). In the simulation we set the occupation probabilities $q = 0.1, 0.87, 0.99, 0.994$ to keep the fraction of giant components $S \approx 0.07$ in all four cases. The plots of simulation results correspond to the average of 100 independent simulation runs. Exact solutions (solid curves) are obtained from Eq. (22).

giant components, the tails of the size distributions of the nongiant components in the case of $L > 0$ are heavier than their counterpart in the case of $L = 0$. This means that, for a fixed fraction of giant components, the sizes of the nongiant components diverge as L grows. In contrast, for Erdos-Renyi graphs with the same fraction of giant components, the tails of the size distributions of the nongiant components in the case of $L > 0$ are lighter than their counterpart in the case of $L = 0$. This means that for a fixed fraction of giant components, the sizes of the nongiant components diverge as L decreases. Precisely explaining this discrepancy is left for future work, which might lead to the characterization of differences between the two popular topologies from the new perspective. It is also worthwhile to note that the error in the case of $L = 1$ is more significant than its counterpart in the other cases, which hints that $L = 1$ is a very critical case (as we have already seen in Fig. 3). Analytically explaining this phenomenon is another interesting future work.

IV. CONCLUSION AND DISCUSSION

We introduced the notion of L-hop percolation, which is of conceptual value, because it captures a sort of *nonindependent* node deletion, and of practical value, because it leads to insights for cybersecurity (e.g., botnet defense). We presented analytic results on the L-hop percolation threshold as well as the mean size and size distribution of nongiant components of complex networks under such operations. In particular, we highlight the following findings: (a) Being able to trace to one-hop neighbors is "almost" as powerful as being able to trace to multihop neighbors, especially in power-law networks. (b) Most nongiant components in the case of Erdos-Renyi random graphs are isolated vertices, whereas typical nongiant components in the case of power-law graphs contain multiple edges. Moreover, as the number of chosen nodes increases, the giant component disappears abruptly in the cases of $L > 0$ but much more smoothly in the case of $L = 0$. (c) The tail of the distribution has a sharp difference between $L = 0$ and $L > 0$ for both Erdos-Renyi random graphs and power-law random graphs when the giant component fraction is kept as a constant.

The notion of L-hop percolation brings a range of interesting problems for future research. In addition to those mentioned in the text, here are more examples. What conclusions we can draw when considering other types of node deletion strategies? If we define some cost functions for node deletions corresponding to $L = 0$ and $L > 0$, respectively, how can we determine the optimal L value? In addition, one reviewer suggested the following interesting research problems: What is the degree distribution of the deleted nodes? Does it strongly differ from that found for $L = 0$? What happens on networks without such strong anticorrelations? What if the selection operations are targeted toward certain nodes? For the power-law degree distributions, is there a kind of power-law distribution when the occupation probability q is very close to the critical point q_c?

ACKNOWLEDGMENTS

We thank Mark Newman for answering our questions regarding our simulation study. We are grateful to the reviewers whose comments helped improve the paper significantly. This work was supported in part by AFOSR Grant No. FA9550-09-1-0165 and by an AFOSR MURI grant.

[1] M. Molloy and B. Reed, Random Struct. Algor. **6**, 161 (1995).
[2] M. Molloy and B. Reed, Comb. Probab. Comput. **7**, 295 (1998).
[3] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. E **64**, 026118 (2001).
[4] N. Fountoulakis, Internet Math. **4**, 329 (2007).
[5] S. Janson, Electron. J. Probab. **14**, 86 (2009).
[6] M. E. J. Newman, Phys. Rev. E **76**, 045101 (2007).
[7] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. Lett. **85**, 5468 (2000).
[8] S. He, S. Li, and H. Ma, Physica A: Statistical Mechanics and Its Applications **388**, 4277 (2009).
[9] J. Grizzard, V. Sharma, C. Nunnery, B. Kang, and D. Dagon, in *Proceedings of the 1st Workshop on Hot Topics in Understanding Botnets* (USENIX Association, Berkeley, 2007), p. 1.
[10] B. Enright, G. Voelker, S. Savage, C. Kanich, and K. Levchenko, USENIX **33**, 2008.
[11] D. Dittrich and S. Dietrich, in *Proceedings of the 3rd International Conference On Malicious and Unwanted Software* (IEEE, Fairfax, 2008), p. 41.
[12] T. Holz, M. Steiner, F. Dahl, E. Biersacky, and F. Freiling, in *Proceedings of the 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats* (USENIX Association, Berkeley, 2008).
[13] G. Sinclair, C. Nunnery, and B. Kang, in *Proceedings of the 4th Annual Conference on Malicious and Unwanted Software* (IEEE, Montreal, 2009), p. 69.
[14] D. Dittrich, USENIX **34**, 35 (2009).
[15] B. Coskun, S. Dietrich, and N. Memon, in *Proceedings of the 26th Annual Computer Security Applications Conference* (ACM, Austin, 2010), p. 131.
[16] B. Stock, J. Göbel, M. Engelberth, F. Freiling, and T. Holz, in *Proceedings of the 2009 European Conference on Computer Network Defense* (IEEE, Washington, 2009), p. 13.
[17] J. Morales, R. Sandhu, and S. Xu, in *Proceedings of the 5th International Conference on Malicious and Unwanted Software* (IEEE, Nancy, 2010), p. 31.
[18] H. Wilf, *Generatingfunctionolgy*, 2nd ed. (Academic Press, London, 1994).
[19] B. Bollobas, *Random Graphs* (Cambridge University Press, New York, 2001).
[20] L. Ahlfors, *Complex Analysis* (McGraw-Hill, New York, 1979).
[21] M. Faloutsos, P. Faloutsos, and C. Faloutsos, in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication* (ACM, New York, 1999), p. 251.
[22] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Rev. Mod. Phys. **80**, 1275 (2008).
[23] M. Newman, SIAM Rev. **45**, 167 (2003).
[24] R. Cohen, S. Havlin, and D. ben-Avraham, Phys. Rev. Lett. **91**, 247901 (2003).