

Network growth with feedback

Raissa M. D'Souza* and Soumen Roy†

*Department of Mechanical and Aeronautical Engineering and Center for Computational Science and Engineering,
University of California, Davis, California 95616, USA*

(Received 21 January 2008; revised manuscript received 19 July 2008; published 27 October 2008)

Existing models of network growth typically have one or two parameters or strategies which are fixed for all times. We introduce a general framework where feedback on the current state of a network is used to dynamically alter the values of such parameters. A specific model is analyzed where limited resources are shared among arriving nodes, all vying to connect close to the root. We show that tunable feedback leads to growth of larger, more efficient networks. Exact results show that linear scaling of resources with system size yields crossover to a trivial condensed state, which can be considerably delayed with sublinear scaling.

DOI: [10.1103/PhysRevE.78.045101](https://doi.org/10.1103/PhysRevE.78.045101)

PACS number(s): 89.75.Fb, 64.60.aq, 02.70.Hm, 89.70.-a

The prevalence and importance of network structures in physical, biological, and social systems is becoming widely recognized. Current research on network growth focuses on models that reproduce aspects of real-world networks, in particular the broad range of node degrees typically observed [1–7]. These simple and elegant models have just one or two free parameters, or strategies that are specified initially and remain unaltered even as the network grows to a massive size, starting from a few seed nodes. Yet the functionality and performance required of a small network may be radically different from that of a large network. Thus, it is natural that the parameters of the growth strategy should change over time as the network grows. The mechanisms underlying these distinct growth models can be generally classified as growth via either preferential attachment [1,3,4], copying [2,6], or optimization [5,7]. In preferential attachment models, the extent of the preference (i.e., the connection kernel) could be altered, tuning properties of the resulting degree distribution [3,8]. In copying models, the probability of successfully copying links could be changed, thus affecting degree distribution. In optimization models, the explicit parameter values of the optimization function could be altered, leading to a range of interesting behaviors [5,7,9].

In this Rapid Communication, we introduce a framework where information on the current state of a network provides *feedback* to the system, allowing it to dynamically alter and self-tune the parameter values throughout the growth process. It combines local optimization models of growth [7,9] with measures of efficient information flow in a network [10]. We show that with feedback one can grow larger and more efficient network structures in less time. This framework can be applied to many systems exhibiting a hierarchical “chain of command” structure. Simple examples are business enterprises, armed forces, etc., with the chief executive officer or the commander in chief, respectively, being the root node of the hierarchy. Such a structure has also recently been found in the organization of genetic regulatory networks [11]. More generally, hierarchy appears to be a central organizing principle of complex networks, providing insight into structures such as food webs and biochemical and social networks [12].

We are interested in growth of hierarchical networks where information flow is essential to the network’s function. Two basic considerations are (i) ensuring a smooth “flow” of commands or information *throughout* the structure, and (ii) addition of new nodes subject to constraints on resources. More explicitly, only some fraction $0 < c \leq 1$ of existing resources can be dedicated to optimizing new growth. The remaining portion of the system is involved with performing some task (e.g, information processing, regulation, transport, and routing), crucial to the sustenance and function of the organization. We show herein that how the resources allocated for growth scale with system size N directly impacts the resulting network structure. Moreover, we show that incorporating feedback leads to flatter hierarchies on which information flows more efficiently, providing a quantitative underpinning to previous case studies of individual organizations where this is found in practice [13].

We consider a simple growth model incorporating (i) and (ii). It is a discrete time process starting from a single root node. Let $G(t)$ denote the network at time t and $N(t)$ the number of nodes. At each time step, an integer number of new nodes, $\lambda(t) \geq 1$, arrive and must connect to the existing network. In accord with (ii), the fraction $0 < c \leq 1$ of resources dedicated to optimizing new growth must be shared equally by all $\lambda(t)$ arriving nodes. Thus, each arriving node sees only $k(t) = [c/\lambda(t)][N(t)]^\alpha$ randomly chosen *candidate* parent nodes, where α determines the scaling of resources and system size, e.g., $\alpha=1$ is linear scaling. It then chooses the one candidate parent that is optimal in some sense (with degeneracy broken by a random choice) and connects to it. Even the simple criterion that optimal is the candidate closest to the root node will demonstrate the importance of feedback.

To elaborate on (i), the efficiency of information flow on $G(t)$ quantifies the network fitness $\mathcal{F}(G(t))$ and is measured by the characteristic time scale τ_c (see below and [10]), for a weighted random walk on $G(t)$. Other measures exist, e.g., [14], yet τ_c is used herein due to its simplicity. $\mathcal{F}(G(t))$ is assessed every δ time steps. If it is found to increase, the system is rewarded by increased arrival rate λ . If it decreases, the system is penalized by decreased λ . Thus, starting from initial value $\lambda_0 \geq 1$, due to *feedback*, $\lambda(t) \equiv \lambda_t$ evolves as

*rmdsouza@ucdavis.edu

†sroy@cse.ucdavis.edu

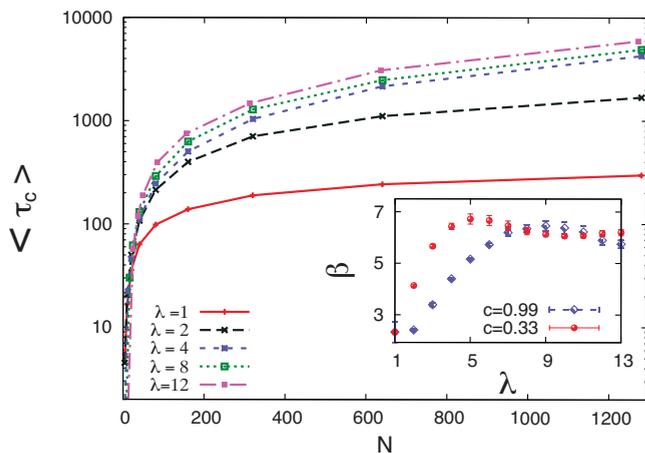


FIG. 1. (Color online) Data points are the expectation value $\langle \tau_c(N) \rangle$ of 1000 independent realizations with $c=1/3$ and varying values of λ as indicated. We find $\langle \tau_c(N) \rangle \sim \ln N^\beta$ where β depends on c and λ as shown in the inset.

$$\lambda_{t+1} = \begin{cases} \lambda_t & \text{if } \mathcal{F}(G(t)) = \mathcal{F}(G(t-\delta)), \\ \lambda_t + 1 & \text{if } \mathcal{F}(G(t)) > \mathcal{F}(G(t-\delta)), \\ \max(\lambda_t - 1, 1) & \text{if } \mathcal{F}(G(t)) < \mathcal{F}(G(t-\delta)). \end{cases} \quad (1)$$

λ is thus a tunable parameter and our goal is to use feedback to tune λ during the growth process to build larger and more efficient structures.

We capture a basic feedback loop [Fig. 3(b), inset]. For a given c , as λ increases, c/λ decreases, generating less efficient structures (i.e., bigger τ_c) which will curtail the growth rate, and vice versa. There is a direct analogy to a business enterprise or an army, where an increase in the rate of employment leads to a smaller portion of resources given to optimizing the attachment of any individual new member. Thus, during rapid growth spurts, hiring is likely less optimal than during periods of slow growth. All techniques used herein are applicable to networks, but for simplicity we consider a tree where each arriving node connects to just one parent.

The characteristic time τ_c is evaluated as in [10], where it was shown to be a performance metric for comparing alternate network topologies. Applications to sensor and to mobile network constructions are discussed in [15–17] and a similar derivation of τ_c is in [18]. A random walk on $G(t)$ is considered, where the walker represents a message to be communicated. We assume unicast communication (i.e., a node exchanges messages with only one other node at a time) and that all nodes constantly attempt to transmit messages. More specifically, if node i is connected to d_i neighbors, it successfully transmits on average $1/d_i$ fraction of the time to one neighbor chosen at random (i.e., with probability $1/d_i$). The remaining $1 - 1/d_i$ fraction of the time transmission is not successful and the message remains on node i , waiting to be transmitted. The state transition matrix P describes this process, where element P_{ij} is the probability of the message passing from node i to j at any discrete time step, with P_{ii} the probability of an unsuccessful attempt. $P_{ij}=0$ if i and j are not directly connected in $G(t)$; otherwise

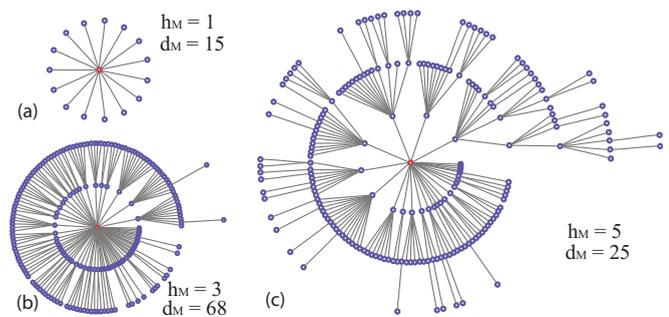


FIG. 2. (Color online) With no feedback networks range from stars (a) to random recursive trees. Example of networks with (a) $c/\lambda=1$, $N=16$, (b) $c/\lambda=1/3$, $N=200$, (c) $c/\lambda=1/10$, $N=200$, with maximum degree d_M and depth h_M indicated.

$$P_{ij} = \begin{cases} 1/d_i^2 & \text{if } i \neq j, \\ 1 - 1/d_i & \text{if } i = j. \end{cases} \quad (2)$$

P is column stochastic and irreducible. Let r_i and \vec{v}_i denote the eigenvalues and eigenvectors of P . By the Perron-Frobenius theorem, there is one eigenvalue $r_1=1$ corresponding to the unique steady-state distribution. All remaining eigenvalues have $|r_i| < 1$ and are modes that decay to the steady state. The characteristic time τ_i for mode i to decay by a factor of $1/e$ is defined by the equality $P^{\tau_i} \vec{v}_i = (r_i)^{\tau_i} \vec{v}_i$ and setting $|r_i|^{\tau_i} = 1/e$. The longest characteristic time τ_c results from r_2 (the largest $r_i < 1$). Rearranging, $\tau_c = -1/\ln|r_2|$.

To implement Eq. (1) we need to compare $\mathcal{F}(G)$ for two networks with different sizes. Yet as N increases, τ_c typically increases. No rigorous results exist describing the relationship. We find empirically that $\langle \tau_c(N) \rangle \sim \ln N^\beta$, where β depends on c and λ , as shown in Fig. 1. Fitness of any particular realization is thus evaluated as $\mathcal{F}(G) = -\tau_c / \ln N^\beta$ (relative to the ensemble of networks with those specific values of c , λ , and N). Note, the negative sign is due to a larger τ_c being less fit.

We analyze the model above via computer simulation, implementing it in R and visualizing results with GRAPHVIZ [19]. First we consider no feedback (λ constant) and linear scaling of $k(t) = \lceil cN(t)/\lambda \rceil$. The notation $\lceil a \rceil$ denotes the closest integer greater than or equal to a , used since k , the number of candidate parents, must be an integer. Linear scaling provides intuition on how c/λ tunes the structures. There are two limiting behaviors: $c/\lambda=1$ (i.e., $k=N$) generates a star topology; and $c/\lambda \rightarrow 0$ (i.e., $k=1$) generates exactly random recursive trees [20]. Figure 2 shows representative networks grown with three different fixed values of c/λ , with maximum node degree d_M and maximum depth h_M indicated.

On incorporating feedback [Eq. (1) with δ finite], λ becomes a tunable parameter. For fixed c , as λ increases the system moves toward $k=1$ (adding new layers of hierarchy). As λ decreases the system moves toward $k=N$ (filling in existing levels of the hierarchy). Thus adjustments in λ tune the levels of hierarchy (and the degree assortativity [21]). Figure 3(a) shows a typical network grown with feedback where $c=1/3$, $\lambda_0=1$, and $\delta=2$, grown to size $N=200$. It has the same initial conditions and final size as in Fig. 2(b); however, with $d_M=17$ it resembles a more balanced version

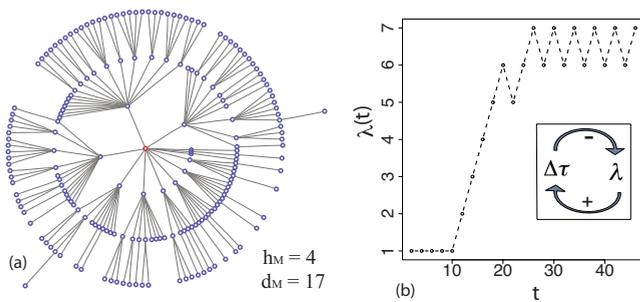


FIG. 3. (Color online) (a) A typical network grown with feedback. Here $c=1/3$, $\lambda_0=1$, $N=200$, and $\delta=2$. This network interpolates dynamically between those in Fig. 2. (b) $\lambda(t)$ for this realization. Inset: schematic of feedback loop.

of Fig. 2(c). Also, the root is no longer the highest-degree node. Figure 3(b) shows the evolution of $\lambda(t)$ for this realization, which is representative of the typical behavior observed (in particular, the final steady-state oscillation).

For larger networks, the consequences of linear scaling between $k(t)$ and $N(t)$ become manifest even in the absence of feedback. With no feedback, using $k(t)=[cN(t)/\lambda(t)]$ quickly leads to a “condensed” structure where all new nodes join with depth $h \leq 2$. Much of the analysis in [9] applies here, except now $k(t)$ is a function rather than a fixed constant. Let $Q_j(t)$ denote the number of nodes at depth j , and $\mu_j(t)$ the expected number of nodes at depth j in the candidate set, $\mu_j(t)=k(t)Q_j(t)/N(t)$. Boundary conditions are $Q_0(t)=1$ for all $t \geq 0$, and $Q_j(0)=0$ for all $j \geq 1$. We can explicitly calculate the exact recurrence $Q_1(t)=\sum_{n=1}^{t-1}k(n)/n$. Approximating the discrete sum by integration, for $k(t)=cN(t)/\lambda$, we find $Q_1(t) \approx cN(t)/\lambda$ and accordingly $\mu_1(t)=(c/\lambda)^2N(t)$. Thus once $N(t) > N_x=(\lambda/c)^2$, $\mu_1(t) > 1$, and with high probability all further incoming nodes join with $h \leq 2$. Figure 4(a) shows this crossover of the depth distribution for fixed $c/\lambda=0.02$ (with crossover length $N_x=2500$).

For sublinear scaling, such as $k(t)=[c\sqrt{N(t)}/\lambda(t)]$, condensation to $h \leq 2$ can be avoided. $Q_1(t) \approx 2c\sqrt{N}/\lambda$, and thus $\mu_1=2(c/\lambda)^2 < 1$ so long as $c/\lambda < 1/\sqrt{2}$ (independent of N). Yet, we eventually see condensation to depth $h \leq 3$ happen once $Q_2(t)$ grows large. Asymptotically $Q_2(t)=\int[k(n)Q_1(n)/n]dn \approx 2c^2N/\lambda^2$. Once $\mu_2(t)=2c^3N^{1/2}/\lambda^3 > 1$, all subsequent nodes join with $h \leq 3$, which occurs at cross-

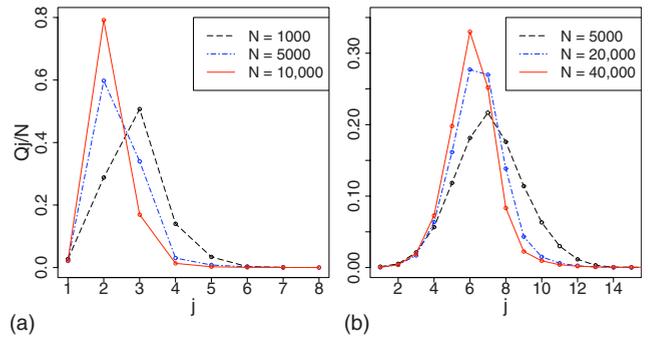


FIG. 4. (Color online) Depth distribution for $c/\lambda=0.02$. (a) Linear scaling, $k \sim N$, results in condensation to $h \leq 2$ for $N > N_x=(\lambda/c)^2$. (b) For $k \sim \sqrt{N}$, the distribution remains concentrated well above the condensed structure for $N \leq N_x=0.25(\lambda/c)^6$.

over length $N_x=0.25(\lambda/c)^6$. Figure 4(b) shows the evolution of the depth distribution for $N \leq N_x$ with $c/\lambda=0.02$ (here $N_x=4 \times 10^9$). It becomes more sharply peaked with increasing N and shifts toward lower average depth but remains concentrated well above the final condensed state. In general, we can show that $k \sim N^{1/a}$, for a any integer, ultimately leads to condensation at depths $h \leq a+1$, with crossover as large as $N_x \sim (\lambda/c)^{a(a+1)}$ [22]. For logarithmic scaling, $k(t)=[c \ln[N(t)]/\lambda(t)]$, the peak of the depth distribution increases as $j=\ln N/\ln \ln N$, and collapse is avoided altogether [22].

Now that the dependence between $k(t)$ and N is understood in the absence of feedback, we can incorporate feedback. We are interested in realistic values of $c \sim 1/3$ and networks of $N \sim 1000$ for which square-root scaling, $k(t)=[c\sqrt{N(t)}/\lambda(t)]$, is sufficient to avoid crossover. We numerically generate ensembles of 100 independent realizations at various values of c and λ_0 , all of which produce similar results. Table I summarizes numerical results for $c=1/3$ and $\lambda_0=3$ (here $N_x \sim 10^5$). Column 2 is the baseline behavior with no feedback. Comparing this with columns 3 and 5 shows that feedback leads to more efficient networks grown to the same size ($N=501$) in less time, with greater depth and lower maximum degree. Comparing column 2 with 4 and 6 shows that, in a given time interval, with feedback, networks grow about twice as large and have improved efficiency. In general, we find that these desirable outcomes are enhanced

TABLE I. Average network properties, over 100 independent realizations, for $c=1/3$, $\lambda_0=3$, and $k(t)=[c\sqrt{N(t)}/\lambda(t)]$. Comparing columns 2, 3, and 5 shows that with feedback networks grow to be of size $N=501$ in much shorter time and are more efficient (smaller $\tau_c/\ln N^\beta$). They also have greater maximum depth and lower maximum degree. Comparing columns 2, 4, and 6 shows that, in a given time interval, networks with feedback grow to be about twice the size and are more efficient.

| | $\delta \rightarrow \infty$ | $\delta=5, N_{\text{stop}}=501$ | $\delta=5, t_{\text{stop}}=167$ | $\delta=1, N_{\text{stop}}=501$ | $\delta=1, t_{\text{stop}}=167$ |
|--------------------------------------|-----------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| $\langle N(t) \rangle$ | 501 | 501 | 956 ± 11 | 501 | 1101 ± 7 |
| $\langle \text{time} \rangle$ | 167 | 89.3 ± 0.5 | 167 | 77.1 ± 0.6 | 167 |
| $\langle h \rangle$ | 3.08 ± 0.02 | 3.41 ± 0.03 | 3.72 ± 0.03 | 3.38 ± 0.02 | 3.76 ± 0.02 |
| $\langle h_M \rangle$ | 6.3 ± 0.7 | 7.09 ± 0.08 | 7.72 ± 0.08 | 7.10 ± 0.08 | 7.82 ± 0.09 |
| $\langle d_M \rangle$ | 32.2 ± 0.6 | 25.2 ± 0.4 | 31.1 ± 0.4 | 24.8 ± 0.4 | 32.7 ± 0.5 |
| $\langle \tau_c/\ln N^\beta \rangle$ | 0.057 ± 0.001 | 0.014 ± 0.001 | 0.014 ± 0.001 | 0.014 ± 0.001 | 0.016 ± 0.001 |

the more often feedback is evaluated. The time required to attain $N=501$ decreases linearly with decreasing δ and the network size attained in an allotted time interval increases linearly with decreasing δ . Of course, each time feedback is assessed resources are needed. In our numerical implementation, they are computational resources. Determining the optimal value of δ would require assessing the trade-off between this increase in resources and the enhanced network properties. For $\delta \leq 10$, our simulations do not show significant sample-to-sample fluctuations in $\langle \mathcal{F}(\mathcal{G}) \rangle$. An exhaustive study of self-averaging in networks [23] with feedback may be discussed elsewhere [22].

In summary, we introduce a general framework for incorporating feedback into network growth models. Proof of concept is demonstrated using a simple model of a hierarchical network where limited resources are shared among all arriving nodes, vying to minimize their distance to the root. Feedback leads to growth of larger, more efficient structures. Linear scaling of resources results in crossover to a trivial condensed structure which can be considerably delayed with sublinear scaling. In the context of a growing organization, this suggests sublinear scaling is necessary once $N \sim 1000$. It may be possible to obtain rigorous results for this model of

network growth with feedback by interpreting c/λ as a branching rate or by proving convergence of $\lambda(t)$ to steady-state oscillation.

The general framework proposed herein allows flexibility in choosing other growth models, communication models between nodes P_{ij} (e.g., broadcast rather than unicast), and fitness functions $\mathcal{F}(G(t))$ (e.g., fitness landscapes [24] modeling random evolutionary pressures). Such alternate choices of $\mathcal{F}(G(t))$ may overcome the current limitation that global topology information is required to assess τ_c . An alternate growth model, where nodes maximize their distance to the root, also seems to demonstrate similar effects of feedback.

Other recent models that could provide mechanisms for introducing feedback are a model of two interacting networks [25], a generative model where loops within a network are considered as potential feedback channels [26], and the layered network framework of [27].

Discussions with C. D'Souza, J. Chayes, C. Borgs, P. Krapivsky, J. Machta, C. Moore and J. Pincus, and held while attending MSRI's "Real World Data Networks" and IPAM's "Random Shapes" workshops are gratefully acknowledged.

-
- [1] A.-L. Barabási and R. Albert, *Science* **286**, 509 (1999).
- [2] J. Kleinberg *et al.*, in *Computing and Combinatorics*, edited by T. Asano, H. Imai, D. T. Lee, S. Nakano, T. Tokuyama, Lecture Notes in Computer Science Vol. 1627 (Springer, Berlin, 1999), pp. 1–17; R. Kumar *et al.*, in *Proceedings of the 41st IEEE Symposium on Foundations of Computer Science*, (IEEE Computer Society, Washington D.C., 2000), pp. 57–65.
- [3] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin, *Phys. Rev. Lett.* **85**, 4633 (2000).
- [4] P. L. Krapivsky and S. Redner, *Phys. Rev. E* **63**, 066123 (2001).
- [5] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou, in *Automata, Languages and Programming*, edited by P. Widmayer, F. Triguero Ruiz, R. Morales, M. Hennessy, S. Eidenbenz and R. Conejo, Lecture Notes in Computer Science Vol. 2380 (Springer, Berlin, 2002), p. 781.
- [6] A. Bhan, D. Galas, and D. G. Dewey, *Bioinformatics* **18**, 1486 (2002).
- [7] R. M. D'Souza, C. Borgs, J. T. Chayes, N. Berger, and R. D. Kleinberg, *Proc. Natl. Acad. Sci. U.S.A.* **104**, 6112 (2007).
- [8] P. L. Krapivsky, S. Redner, and F. Leyvraz, *Phys. Rev. Lett.* **85**, 4629 (2000); C. Cooper and A. Frieze, *Random Struct. Algorithms* **22**, 311(2003); P. G. Buckley and D. Osthus, *Discrete Math.* **282**, 53 (2004).
- [9] R. M. D'Souza, P. L. Krapivsky, and C. Moore, *Eur. Phys. J. B* **59**, 535 (2007).
- [10] R. M. D'Souza, S. Ramanathan, and D. Temple Lang, in *Proceedings of IEEE INFOCOM* (IEEE Computer Society, Washington, D.C., 2003), p. 1564.
- [11] H. Yu and M. Gerstein, *Proc. Natl. Acad. Sci. U.S.A.* **103**, 14724 (2006).
- [12] A. Clauset, C. Moore, and M. E. J. Newman, *Nature* (London) **453**, 98 (2008); S. Redner, *ibid.* **453**, 47 (2008).
- [13] C. R. James, *Organizational Dynamics* **32**, 46 (2003); P. Iskanian, H. Helaakoski, A. M. Alaruikka, and J. Kipina, in *Proceedings of the IEEE Conference on Engineering Management* (IEEE Computer Society, Washington, D.C., 2004).
- [14] P. Gupta and P. R. Kumar, *IEEE Trans. Inf. Theory* **46**, 388 (2000).
- [15] H. Koskinen, in *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, edited by S. Balsamo, C. -F. Chiasserini and L. Donatiello (ACM Press, New York, 2004), p. 1.
- [16] R. Badonnel, R. State, and O. Festor, *Telecommun. Syst.* **30**(1–3), 143 (2005).
- [17] S. Pandey and P. Agrawal, In *Proc. IEEE Global Telecom. Conf.* (2006).
- [18] J. D. Noh and H. Rieger, *Phys. Rev. Lett.* **92**, 118701 (2004).
- [19] The R Foundation for Statistical Computing, www.R-project.org/; AT&T and Pixelglow Software, www.pixelglow.com/graphviz/
- [20] R. T. Smythe and H. Mahmoud, *Theory Probab. Math Statist.* **51**, 1–29 (1996).
- [21] M. E. J. Newman, *Phys. Rev. Lett.* **89**, 208701 (2002).
- [22] R. M. D'Souza and P. L. Krapivsky (unpublished).
- [23] S. Roy and S. M. Bhattacharjee, *Phys. Lett. A* **352**, 13 (2006).
- [24] M. Mitchell, *An Introduction to Genetic Algorithms* (MIT Press, Cambridge, MA, 1996).
- [25] K. Börner, J. T. Maru, and R. L. Goldstone, *Proc. Natl. Acad. Sci. U.S.A.* **101**, 5266 (2004).
- [26] D. R. White, N. Kejžar, C. Tsallis, D. Farmer, and S. White, *Phys. Rev. E* **73**, 016119 (2006).
- [27] M. Kurant and P. Thiran, *Phys. Rev. Lett.* **96**, 138701 (2006).