Self-organized criticality in a computer network model

Jian Yuan,* Yong Ren, and Xiuming Shan
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
(Received 6 May 1999)

We study the collective behavior of computer network nodes by using a cellular automaton model. The results show that when the load of network is constant, the throughputs and buffer contents of nodes are power-law distributed in both space and time. Also the feature of $1/f$ noise appears in the power spectrum of the change of the number of nodes that bear a fixed part of the system load. It can be seen as yet another example of self-organized criticality. Power-law decay in the distribution of buffer contents implies that heavy network congestion occurs with small probability. The temporal power-law distribution for throughput might be a reasonable explanation for the observed self-similarity in computer network traffic.

PACS number(s): 05.40.−a, 89.80.+h

I. INTRODUCTION

The rapidly growing global data network (Internet), connecting millions of computers, has gradually reached a considerably huge scale for which methods of statistical physics, especially of complex systems, can play a role in analysis. The whole Internet can be considered as a single system. The general picture of this system has recently been unveiled by researchers at the Cooperative Association for Internet Data Analysis (CAIDA) in San Diego, California [1]. The topology of the system is irregular. It has a hierarchical, treelike structure, with some loops. At the vortices of the system there are many nodes, called gateways, routers, bridges, etc., which help to route the data packets to the destination, and have significant autonomy in finding the optimal way from source to destination.

Simulating how the Internet behaves is an immensely challenging undertaking because of its heterogeneity and rapid change. The heterogeneity ranges from the individual links that carry the network traffic to the protocols that interoperate over the links, and to the ''mix'' of different applications used at a site and the levers of congestion (load) seen on different links [2]. The design of the Internet continues to evolve, and many aspects of its behavior are poorly understood. Due to the network complexity, simulation plays a crucial role in attempting to characterize how different facets of the network behave, and how proposed changes might affect the network properties. Yet simulating different aspects of the Internet is exceedingly difficult. Thus one key strategy for developing meaningful simulations in the face of these difficulties is searching for invariability. By the term invariability we mean facets of network behavior which have been empirically shown to hold in a very wide range of environments.

A number of recent empirical studies on traffic measurements from a variety of working packet networks have convincingly demonstrated that actual network traffic is self-similar in nature. However, the reasons behind network traffic self-similarity have not been clearly identified. The work by Willinger *et al.*, provides a simple explanation [3]. They show that the self-similar traffic can be constructed by multiplexing a large number of ON/OFF sources that have heavy-tailed ON and OFF period length. Some interrelated work involved the effects of file sizes, transport protocols [4], and even user behaviors [5]. However, this explanation obviously ignores the nonlinearity arising from the interaction of traffic sources competing for network resources. Moreover, it does not stress the interaction of autonomous nodes in this driven system, which can produce collective phenomena.

When the load offered to any network is more than it can handle, congestion builds up. The Internet is no exception. In theory, congestion can be dealt with by employing a principle borrowed from physics: the law of conservation of packets, by which a new packet will not be injected into the network until an old one leaves. TCP (transmission control protocol) in the Internet attempts to achieve this goal by dynamically manipulating the source rates [6]. In addition, there is not any cooperation between the nodes. The Internet users still encounter congestion frequently, which may emerge at different degrees, and last different periods. The effect of aggravating load and TCP can be easily understood in a bottleneck. However, fathoming the congestion varying spatiotemporally is really a hard question.

Until recently, the researches of Internet dynamics have mainly concentrated on two facets in the network community. On the one hand, the investigators study dynamics of one single node in order to understand the dynamics of the whole network. They believe that they can predict the behavior of a large interactive system by studying its elements separately and by analyzing its microscopic mechanism individually. On the other hand, they pay attention to influences from outside the Internet. For example, Huberman and Lukose stated that because most users pay a flat rate for unlimited access, user behavior causes the Internet's congestion [7]. Park *et al.* reckoned that self-similar traffic is induced by heavy-tailed distribution of file size [4]. Obviously, the Internet is an interactive dissipative dynamical system. Complex behaviors may still emerge in the Internet with simple or even changeless load. However, for lack of a better theory, there has not been much literature on the dynamics of the Internet itself.

*FAX: +86 010 62781382.
Electronic address: yuanjn@sdp.ee.tsinghua.edu.cn

We cannot predict when and where the congestion occurs. Such a plight is very similar to what we meet in highway traffic. Recently cellular automaton models have been increasingly used in simulations of highway traffic [8,9]. Physicists have contributed a lot to the better understanding of traffic flow. The methods of statistical physics and nonlinear dynamics have been successfully applied to these models, stressing the notion of a dynamical transition from low-density laminar flow to high-density jammed behavior [10]. Csabai called researchers' attention to the close analogy between the basics of data traffic on the Internet and vehicle traffic on the highway a few years ago [11]. His measurement also showed that the power spectrum of the round-trip time between two points of the Internet is $1/f$-like. Although he is not continuing his research work, he still pointed out that the models of collective phenomena, such as highway traffic models, could be appropriate for describing the behavior of data communication in computer networks. On the other hand, some researchers used the concept of phase transition to explain the fluctuation of packet delivery time in computer networks. They found that the network operates most efficiently in the vicinity of the critical point, and inferred that the shifting of the phase transition point means that the collective behavior of routers may play a crucial role in deciding the congestion nature of the network [12,13]. Takayasu *et al.* theoretically clarified the phase transition behavior of buffer dynamics [14]. They proposed a simple dynamical phase transition model, the contact process, to explain $1/f$-type noise in round trip time with the Cayley tree (or Bethe lattice) [15], and also demonstrated the existence of strong spatial correlation of congestion levels [16]. In our other paper, we studied a simple model at the packet level to describe the outflow from a packet jam self-organizes to a critical state of maximum throughput [17]. We showed that the emergence of collective behavior of packets causes the long-range dependence of congestion in the computer networks.

In this paper, we study the dynamical principles of computer networks at node level. We also attempt to show that the emergent phenomena from a collection of autonomous nodes which adaptively self-organize into a complex state with ubiquitous power laws as the classic hallmarks of criticality, can result in the traffic self-similarity and the unpredictable congestion in the Internet.

The outline of this paper is as follows. We describe the model in Sec. II, and discuss its phenomenological behaviors and simulation results in Sec. III. Section IV gives a short discussion.

II. MODEL

A typical bottleneck node model is depicted in Fig. 1(a). In this model, some TCP controlled sources share a node which has a buffer and a server with the service rate μ. When the node receives more packets sent by these sources than it can deal with, it momentarily queues the packets not forwarded immediately in its buffer, thus increasing the delay of the packets through the network. This is similar to the case when many vehicles crowd into a section of a highway, where the traffic self-organizes to a critical state of maximum outflow [9]. This feature of maximum throughput selection is characteristic of a driven diffusive system, and the critical state is the most efficient state that can be actually reached dynamically [18,19]. Therefore a node could operate most efficiently in the vicinity of a critical state with maximum throughput and the least packet numbers in its buffer. Let us consider now a heavily loaded computer network to study the collective behavior of the nodes which are the interaction agents below, not taking into account the mutual interaction of the packets themselves. The interaction of nodes can produce collective phenomena, which cannot be foreseen from examination of the counterparts.

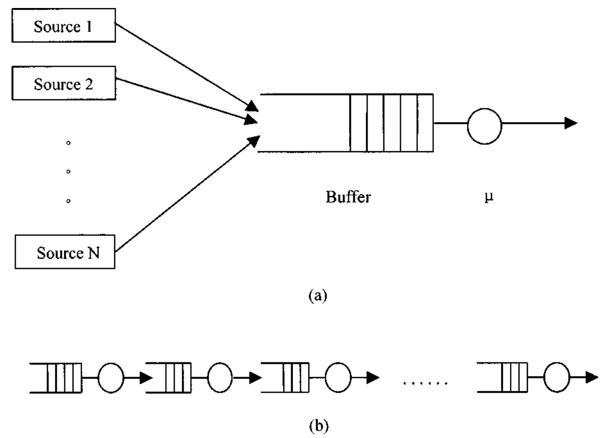

FIG. 1. (a) A bottleneck node, which has a buffer and a server with the service rate μ, shared by some sources controlled by TCP. (b) Many nodes are linked in series, all the side branches of network are discarded, and only the line of a two-point connection is described.

We first mention Kleinrock's independence assumption, which dominates over the network analysis [20]. With this assumption, the input processes of all nodes are independent of each other, and then the network analysis is not too difficult. Actually, a series of packets entering a node will be the input of the next node. Two input processes are obviously correlated in time and quantity. In order to illustrate this correlation, we link the nodes, discarding all the side branches of network, and only describe the line of a two-point connection. This setup is depicted in Fig. 1(b). So we could use a one-dimensional cellular automaton model to simulate the behavior of network nodes. To our knowledge, this paper might be the first to suggest such a model for a computer network.

Our computational model is defined on a one-dimensional array of L sites with open or periodic boundary conditions. Each site may be empty or occupied by one particle. Let M particles be randomly placed in L sites $(L>M)$. Each particle corresponds to a node, and moves from left to right with an integer velocity. The number of empty sites between two contiguous particles represents the buffer contents of the left node. Note that the order of nodes here is in reverse order compared with the order in Fig. 1(b). If a particle runs fast (corresponding to a node with large throughput), the left particle must also move fast in order to reduce the distance to its right one (corresponding to a reduction of the queue length of the left node).

In this paper, we choose the periodic boundary condition, i.e., the output of the left most node being the input of the right most, to keep the network load constant. For an arbi-

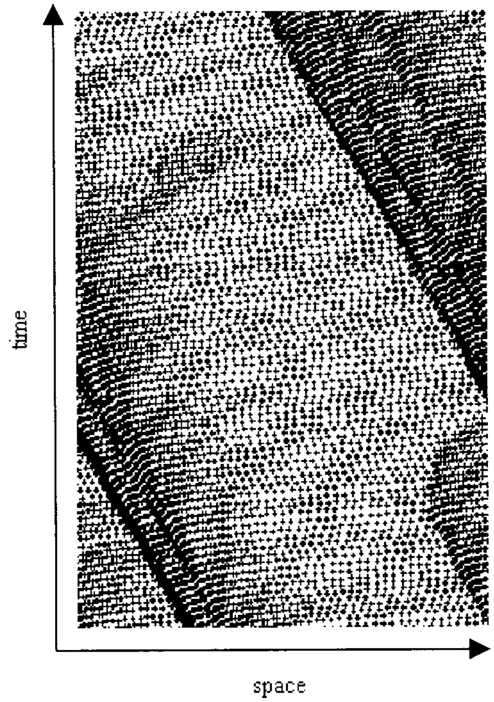

FIG. 2. A typical pattern for a system of size $L=1000$, where $M=100$, $\nu_{max}=20$, and $p=0.05$. 1000 time steps of system evolution are shown. The horizontal direction is space and the vertical direction (up) is (increasing) time.

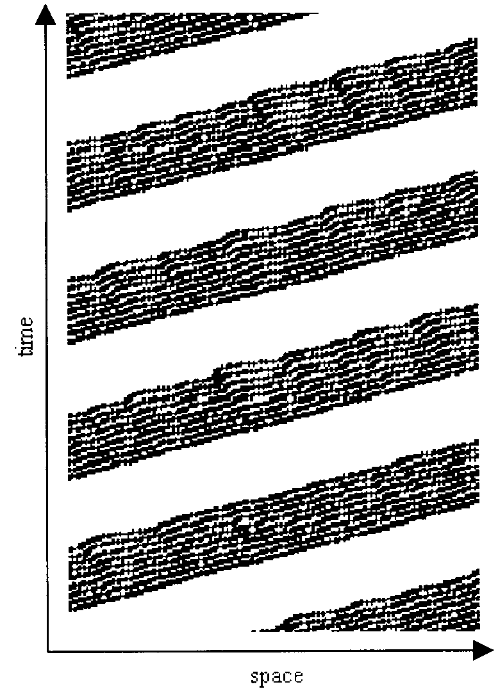

FIG. 3. A bottleneck situation with the same parameters as Fig. 2, except that $\nu_{max}=5$ for one of the nodes.

trary configuration, one update of the system consists of the following steps, which are performed in parallel for all nodes.

(1) If the throughput $\nu_i(n)$ of node i $(1\leqslant i\leqslant M)$ at time n is lower than ν_{max} and if the queue length of node i, $d_i(n)$, is larger than $\nu_i(n)+1$, the throughput is increased by 1, i.e., $\nu_i(n+1)=\nu_i(n)+1$.

(2) If node i presents that $\nu_i(n)\geqslant d_i(n)$, then $\nu_i(n+1)=d_i(n)$.

(3) With probability p, the throughput of each node (if greater than zero) is decreased by 1, i.e., $\nu_i(n+1)=\nu_i(n)-1$.

(4) Each node update with $\nu_i(n+1)$, and then $d_i(n+1)=d_i(n)+\nu_{i+1}(n+1)-\nu_i(n+1)$. For the end most node $i=M$, the node $i+1$ in this step represents node 1.

The third step is essential for simulating the realistic network node since otherwise the dynamics is completely deterministic. It takes into account the natural throughput fluctuation due to the overhead on computer networks, or due to the variation of the external conditions, say, some traffic across the nodes coming from side branches of the network.

III. SIMULATIONS

We first discuss the phenomenological behaviors of the model. A typical pattern is shown in Fig. 2 for a system of size $L=1000$, where $M=100$, $\nu_{max}=20$, and $p=0.05$. Starting with a random initial condition and after discarding a transient period of 5000 iterations, Fig. 2 shows the 1000 time steps of system evolution. The horizontal direction is space and the vertical direction (up) is (increasing) time. Every dot curve exhibits moving process of a particle. For the convenience of observation, only some particles are drawn in the figure instead of all M particles. The smoother the curve is, the larger the throughput of the node. The falling throughputs spread backward as time goes on. The change of the space between particles at different time steps shows the fluctuation of the queue length of the nodes.

In Fig. 3 we show a bottleneck situation with $\nu_{max}=5$ for one node and with the other parameters identical to those in Fig. 2. The buffer contents of the bottleneck node always stay larger than the others. The node always relaxes the congestion with its maximum throughput. The throughputs of other nodes fluctuate around the throughput of the bottleneck node.

To gain more insight into the collective behavior of the nodes, we change the simulation parameters: the system size $L=50\,000$, the maximum throughput of each node $\nu_{max}=100$, the number of nodes $M=4000$, and $p=0.05$. We measure the probability distributions of the throughputs and buffer contents of nodes up to 100 000 time steps. The distribution for queue length d of all nodes at a time step $P_{d\mathrm{space}}(d)$ vs d, is shown in Fig. 4(a), and the distribution for queue length of a single node for all time steps $P_{d\mathrm{time}}(d)$ vs d, is shown in Fig. 4(b). Figure 5 shows the distribution of the throughputs ν of all nodes at a time step $P_{\nu\mathrm{space}}(\nu)$ vs ν [Fig. 5(a)] and of a single node for all time steps $P_{\nu\mathrm{time}}(\nu)$ vs ν [Fig. 5(b)]. The straight lines in Figs. 4(a) and 5(a) have slopes -1.25 and -1.15 in Figs. 4(b) and 5(b). It is seen that the probability distributions for throughputs and buffer contents of nodes are power law in both space and time. The fact that the distributions in Figs. 4 and 5 begin to deviate from power law at large sizes is a finite-size effect.

The queue length can reflect the levers of congestion in nodes. Power-law decay in the distribution of buffer contents implies that heavy congestion in the network occurs with small probability. The temporal power-law distribution for throughput might be a reasonable explanation for the ob-

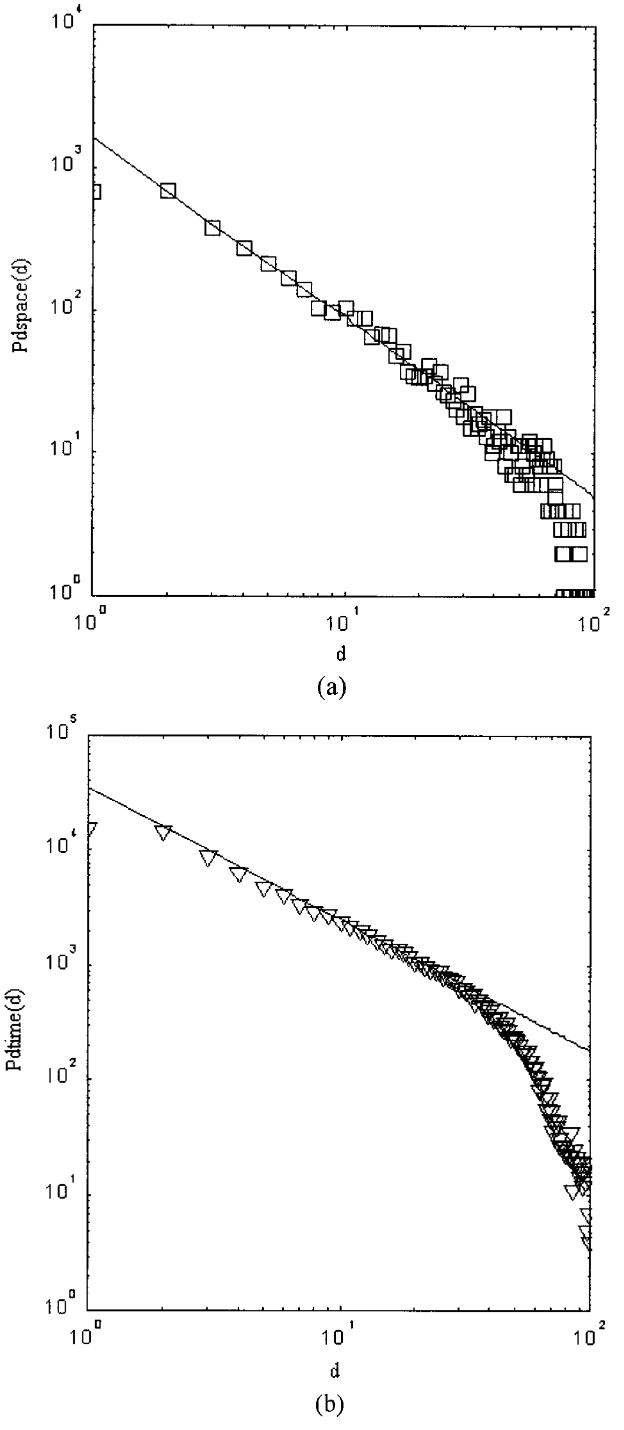

FIG. 4. (a) The distribution for the queue length of all nodes at a time step $P_{d\mathrm{space}}(d)$ vs d and (b) the distribution for the queue length of a single node for all time steps $P_{d\mathrm{time}}(d)$ vs d, for $L=50\,000$, $M=4000$, $\nu_{\max}=100$, and $p=0.05$.

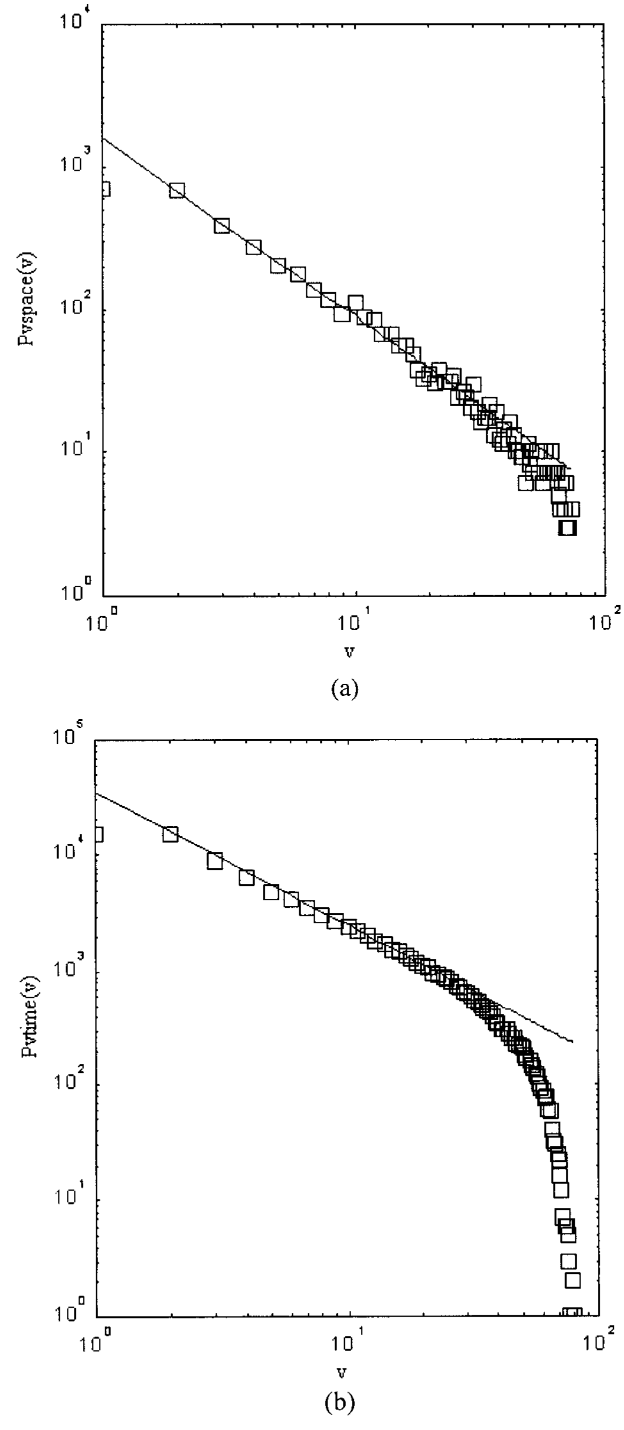

FIG. 5. (a) The distribution for throughputs of all nodes at a time step $P_{\nu\mathrm{space}}(\nu)$ vs ν and (b) the distribution for throughput of a single node for all time steps $P_{\nu\mathrm{time}}(\nu)$ vs ν.

served self-similarity in computer network traffic. To gain more information about the nodes bearing a fixed part of the system load, we first record the time series for the number of nodes $N_l(t)$ in a small segment of length l in the system, and then calculate the power spectrum $S(f)$ of $N_l(t)$ for $l=500$ as shown in Fig. 6. The curve appears as the feature of $1/f$ noise, and has a slope of about -2.3. Therefore, the system exhibits a self-organized criticality. The spatiotemporal scaling in the self-organized critical state does not necessarily manifest itself in nontrivial exponents of the power spectrum. The exponent of the power spectrum depends on the level of conservation [21]. Our current understanding only tells us to expect the power law without nailing down the exponents.

IV. DISCUSSION

Simulating different aspects of the Internet is exceedingly difficult [2]. We do not address the more detailed issues here.

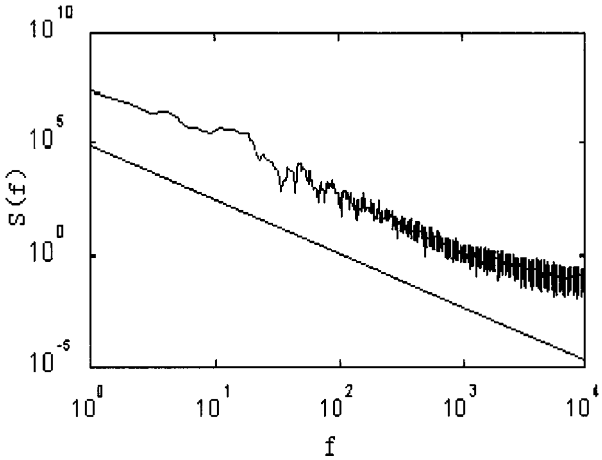

FIG. 6. The time series for the number of nodes $N_l(t)$ in a small segment of length l in the system, was recorded, and then the power spectrum, $S(f)$ of $N_l(t)$ was calculated for $l=500$. The straight line has a slope of -2.3.

Instead, our intent has been to discuss some of the more fundamental issues. The cellular automaton model presented in this paper might provide an approach for analyzing the collective behavior of network nodes.

Because composite systems, such as the Internet, contain many components and are governed by many interactions, analysts cannot possibly construct mathematical models that are both totally realistic and theoretically manageable. They therefore have to resort to simple, idealistic models that capture the essential features of real systems. Cellular automaton models [22] are increasingly used in simulations of complex physical systems such as fluid dynamics [23], sand piles [18], and highway traffic [8,10]. In some of the systems, cellular automaton models provide only some general qualitative features of the system while in other cases useful quantitative information can be obtained. One-dimensional (1D) cellular automaton models are very simple. They are successfully used for capturing the critical slope of sand piles, although the 1D situation looks completely unrealistic for modeling sand piles. 1D models are also unrealistic for computer networks. However, based on such 1D models, it is still feasible to capture the essential features of a network: the network self-organizes into a critical state by interactions of nodes. For selecting the periodic boundary condition in our model, we first consider that the law of conservation of packets in the transmission control protocol of the Internet. Secondly, the emergence of self-organized criticality does not lie on the open boundary condition or the periodic one in the model [24,25], though the former is more reasonable. Third, it is not easy to select a typical open boundary condition. It is convenient for simulation to choose a periodic boundary condition. Of course, it is not quite perfect for both the model and the condition, but this is an initial work in a new area. We hope our work will lead to widespread discussion and more reasonable models for dynamics of the Internet itself.

[1] K. Claffy, T. E. Monk, and D. Mcrobb, Nature (London) **397**, (1999) (web matter: URL http://helix.nature.com/webmatters/); Science **282**, 375 (1998).
[2] V. Paxson and S. Floyd (unpublished).
[3] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, IEEE/ACM Trans. Network **5**, 71 (1997).
[4] K. Park, G. Kim, and M. Crovella (unpublished).
[5] M. E. Crovella, A. Bestavros, and M. S. Taqqu, *A Practical Guide to Heavy Tails: Statistical Techniques and Applications* (Birkhäuser, Boston, 1998).
[6] V. Jacobson (unpublished).
[7] B. A. Huberman and R. M. Lukose, Science **277**, 535 (1997).
[8] P. M. Siman and K. Nagel, Phys. Rev. E **58**, 1286 (1998).
[9] K. Nagel and M. Paczuski, Phys. Rev. E **51**, 2909 (1995).
[10] D. Helbing and M. Treiber, Science **282**, 2001 (1998).
[11] I. Csabai, J. Phys. A **27**, L417 (1994).
[12] A. Y. Tretyakov, H. Takayasu, and M. Takayasu, Physica A **253**, 315 (1998).
[13] T. Ohira and R. Sawatari, Phys. Rev. E **58**, 193 (1998).
[14] M. Takayasu, A. Y. Tretyakov, K. Fukuda, and H. Takayasu (unpublished).
[15] M. Takayasu and T. Sato, Physica A **233**, 824 (1996).
[16] K. Fukuda, M. Takayasu, and H. Takayasu, Fractals **7**, 23 (1999).
[17] J. Yuan, Y. Ren, and X. M. Shan (unpublished).
[18] P. Bak, C. Tong, and K. Wiesenfeld, Phys. Rev. A **38**, 364 (1988).
[19] P. Bak, How Nature Works (Springer-Verlag, New York, 1996).
[20] L. Kleinrock, *Queueing System: Computer Applications* (Wiley, New York, 1976).
[21] K. Christensen, Z. Olami, and P. Bak, Phys. Rev. Lett. **68**, 2417 (1992).
[22] S. Wolfram, *Theory and Applications of Cellular Automata* (World Scientific, Singapore, 1986).
[23] G. D. Doolen, Physica D **47**, 1 (1991).
[24] K. Nagel and M. Schreckenberg, J. Phys. I **2**, 2221 (1992).
[25] T. Nagatani, J. Phys. A **28**, L119 (1995).