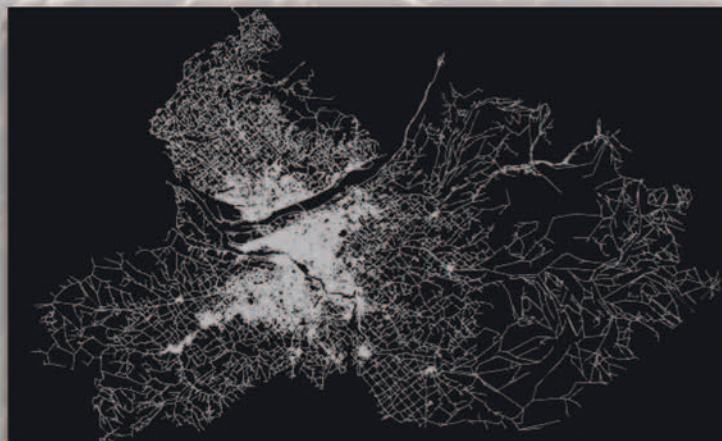


Complex Networks

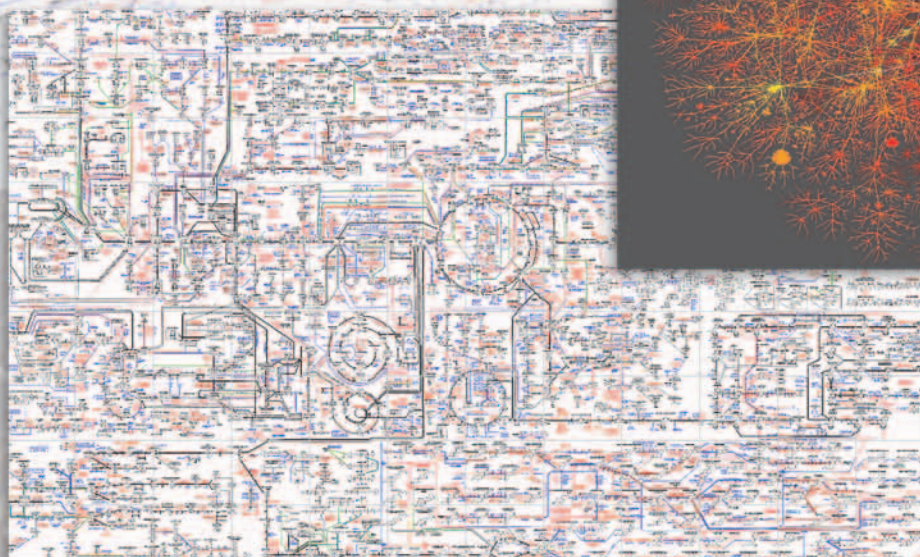
The Challenge of Interaction Topology

Zoltán Toroczkai

Networks have recently become a paradigmatic way of representing complex systems in which the pattern of interactions between a system's constituent parts is itself complex and is evolving together with the system's dynamics. Transport is the main function of these dynamic networks. It is therefore crucial that we understand the coupling between the network structure and the efficiency and robustness of the transport processes on the structure. Such understanding will have a huge impact, allowing us to control signaling processes in the cell and to design robust information and energy-transmission infrastructures, such as the Internet or the power grid. However, achieving this type of understanding is rather challenging, because of the discrete and random nature of network topology. This article reports on some of our results that connect network dynamics and transport efficiency. It also illustrates the power behind the ability to control the topology of the interactions in the design of scalable computer networks.



The roadways of Portland, Oregon.



Macroscopic snapshot of Internet connectivity (skitter data) with selected backbone Internet service providers. (This photo is courtesy of the Lumeta Corporation.)

The metabolic pathway. (Gerhard Michal: *Biochemical Pathways*, 1999 © Elsevier GmbH, Spektrum Akademischer Verlag, Heidelberg.)

Systems of many interacting particles typically exhibit complex behavior. In most well-known complex systems, the topology of the interactions between particles can be described by simple structures, such as regular crystalline lattices or a continuum, and the complex behavior arises from nonlinearity and nonlocality, which describe the nature of the interactions themselves. There is, however, a large class of systems called complex networks, in which the interactions are mediated not by a continuum (or a simple regular structure) but by a complex graph, whose structure may evolve as part of the dynamics of the interactions.

Familiar systems in almost every area of life form such complex networks. Here are a few examples: transport and transportation infrastructures (electric power grids, waterways, natural gas pipelines, roadways,

airlines, and others) social interactions (acquaintance networks, scientific collaboration networks, terrorist networks, sex webs, and others), communications networks (the World Wide Web, the Internet, microwave backbone, and telephone networks), biological networks (metabolic networks, gene regulatory networks, protein interaction networks), and networks in ecology (food webs). Although these systems have been known for a while, their complexity has been explored only recently because the large databases and the immense computational power required to analyze network data were almost nonexistent two decades ago.

Even a cursory “look” at the structure of real-world networks creates a breathtaking impression: These are large objects containing thousands, or sometimes, even hundreds of millions, of nodes with an intricate mesh

of connections among them. For the last decade, the science of complex networks has focused on describing the structural complexity of real-world network topologies. By looking at the three images on these opening pages, one can easily surmise, that statistical and probabilistic methods are essential to that description. Today, the focus has expanded beyond network structure to an understanding of the relationship between structure and dynamics and the implications of that relationship for network design. The first half of this article traces the main ideas in graph theory over the past two centuries, which are at the basis of the mathematical approach to networks, and the second half is devoted to some very recent developments: computer network design and the connection between network dynamics and structure.

The Problem of the Königsberg Bridges

Network images can be quite striking. But one might question whether thinking about complex systems in terms of networks leads to more than pretty pictures. Ironically, the fundamentals of the theory of network structures were introduced by a blind mathematician.

It all began with the puzzle of seven bridges, an entertaining brain-teaser for people who strolled through Königsberg, the Prussian city at the Baltic Sea, in the 18th century. The river Pregel divides the city into four land areas connected by seven bridges. The burghers of Königsberg wondered if one could visit all the four areas by crossing each bridge exactly once (see Figure 1).

The puzzle was solved in 1736 by Leonhard Euler, who at the time, was a mathematics professor in St. Petersburg. The power of Euler's solution lies not in the answer itself (which is negative) but in the way it was derived. Euler's revolutionary idea was to represent the pieces of land separated by bridges as the nodes (dots) *A*, *B*, *C*, and *D* and to represent the bridges as the edges (line segments) *a*, *b*, *c*, *d*, *e*, *f*, and *g*, connecting the nodes (see Figure 2). The structure formed by the set of nodes and edges, called a graph, is a simplified representation of the puzzle, encoding the relationships between the pieces of land and the paths of access between them (see Figure 2 inset). In this representation, the problem translates into the following one: Find a path that visits all nodes but passes through all edges exactly once. Obviously, the intermediate nodes must have an even number of incident edges (if one visits an intermediate node, one must also leave it).

Because the Königsberg puzzle has 4 (>2) nodes, all with an odd number of edges, there can be no such path.

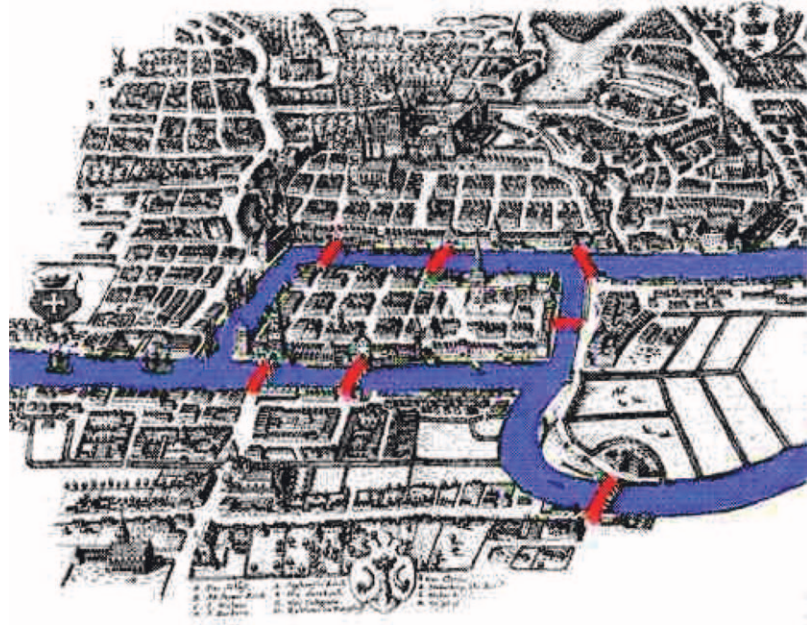


Figure 1. The Königsberg Puzzle

This woodcut shows the ancient Prussian city of Königsberg (now known as Kaliningrad) with its seven bridges across the river Pregel. The possibility of strolling across the city by crossing each bridge once only became the object of a famous brain-teaser.

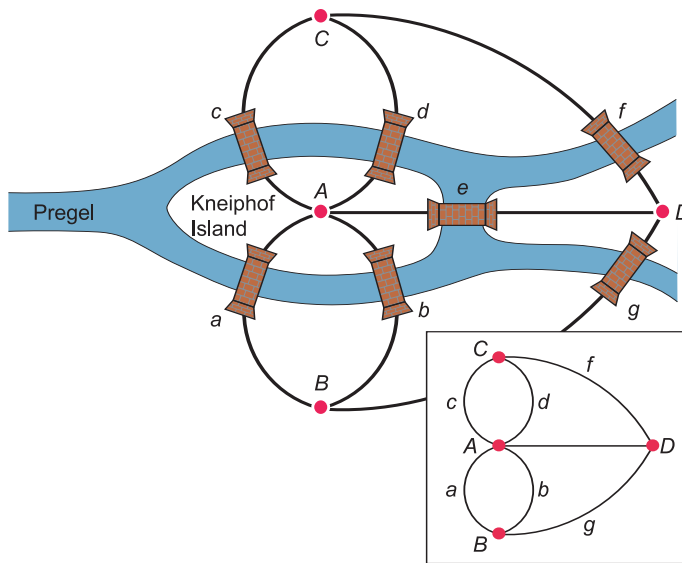


Figure 2. Euler's Solution to the Königsberg Puzzle

A simple representation of the problem by a graph helps realize that there is no such path that visits all nodes and passes through all seven edges exactly once.

Euler's representation of the relationships between a discrete set of entities as a graph led to the development of a particular type of mathematical nomenclature and ultimately to a new field of discrete mathematics called graph theory.

A Hard Problem: The Ramsey Numbers

For nearly 200 years, graph theory was concerned with topological and/or geometrical properties of small structures, or regular structures (such as a lattice). Then, the 1951 seminal paper by Ron Solomonoff and Anatol Rapoport (1951) and the 1959–1960 series of papers by Pál Erdős and Alfréd Rényi caused the rebirth of graph theory. These papers introduced the notion of a random graph and, more important, that of graph ensembles, which are sets of graphs that share a given property Q . To understand this notion, let us look briefly at the famous Party Problem and the Ramsey numbers. This problem, inspired from social interactions, is stated very simply:

What is the minimum number of guests, R , one should invite to a party that would surely have k people who all know each other or k who do not know each other (at all)?

For $k = 3$, it is easy to prove that $R(3) = 6$. We will use Euler's method: Let us denote the six people by the nodes A, B, C, D, E , and F . Let us represent the fact that two people know each other by drawing a red link (or edge) between them and use a blue edge to link two people who do not know each other. Since pairs of people either know each other or do not, the graph obtained is complete, which means that all possible edges are drawn—see Figure 3(a). Specifically, a complete graph with n nodes, denoted here by K_n , always has $n(n - 1)/2$ edges. The graph theoretic

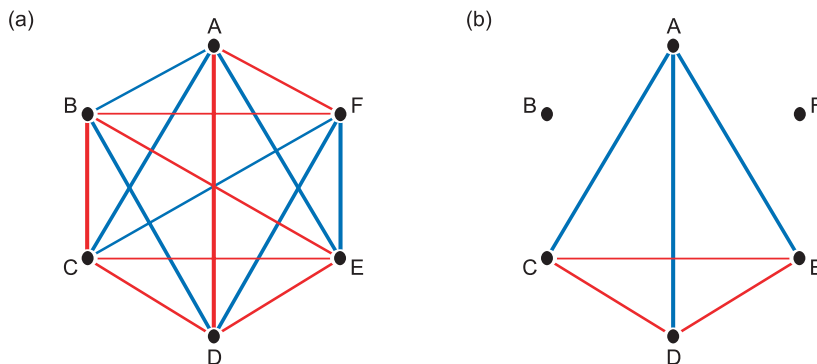


Figure 3. The Party Problem for $k = 3$

Six is the minimum number of people that always contains a group of three, all of whom either know each other (red links) or do not know each other (blue links). (a) A complete graph for six people is shown. Note that it contains the complete subgraph CDE . (b) This figure demonstrates that there is no way to draw a complete graph without constructing a complete single-color three-node subgraph within it. Suppose that blue links indicate that A does not know C, D , or E . In that case, CD, DE , and EC must be red links so that a complete blue subgraph should not be formed. But then, as shown in (b), those three red links form a complete subgraph, which means that C, D , and E know each other.



Leonhard Euler

Leonhard Euler, the most prolific mathematician of all times, was born in Switzerland in 1707 and spent his life in Berlin and St. Petersburg. *Opera Omnia* is an incomplete collection of his works that has 73 volumes, each over 600 pages in length.

version of the Party Problem is thus to determine the minimum number of nodes n , such that a complete graph with n nodes and with edges of two color always has at least one complete subgraph of k nodes with all edges of the same color. For $k = 3$, a complete subgraph is a monochromatic triangle.

If there are $n = 5$ people present, one can easily color a complete graph with no such triangle present, ruling out $n = 5$ by inspection.

For $n = 6$, however, there is always at least one such triangle. To prove this statement, let us assume the opposite, namely, that there can be no such triangles. Since for every node there are $n - 1 = 5$ incident edges but only two colors, there must be 3 edges of the same color incident on the node. For example, consider the edges AC, AD , and AE in Figure 3(b) to be the same color, for example, blue. Since the triangles ACE, ACD , and ADE cannot have all three of their edges of the same color, CE, CD , and DE must be red. Then CDE is a triangle all with the same color edges (red), a contradiction. Hence, $R(3) = 6$.

For $k = 4$, the answer is $R(4) = 18$, which is hard to prove. For $k = 5$ and higher, the answers are not known; only some bounds exist. Although we have no proof for $k = 5$, one might think that we would surely be able to use today's supercomputers to find the value of $R(5)$. However, as Bollobás,

an eminent graph theoretician, has stated (1998) "...a head-on attack by computers for $R(5)$ is doomed to failure" This failure is largely due to the combinatorial explosion in the number of ways we can draw a complete graph with n nodes using two colors for the edges: On the face of it, a computer would have to search a total of $2^{n(n-1)/2}$ such graphs for complete subgraphs with k nodes. For $k = 3$, when $n = R(3) = 6$, there are $2^{15} = 32,768$ complete graphs, for $k = 4$, the analytic solution gives $n = 18$, which means that there are 2^{153} , or approximately 1.46×10^{46} graphs. For $k = 5$, the best known bounds are $43 \leq R(5) \leq 49$, which would mean approximately 2^{903} to 2^{1176} graphs (or on the order of 10^{301} graphs). For $k = 5$ and $n = 43$, the "ultimate laptop" of Seth Lloyd (2000), which operates at the physical limit of computation (as determined by the speed of light, the Planck constant, and the gravitational constant), performing $f = 5.4258 \times 10^{50}$ operations per second, would have to work for at least 2.693×10^{213} years, a mighty long time (the age of the universe is estimated to be between 1.1×10^9 and 2×10^9 years).

So, can we hope ever to solve the Party Problem? The key idea is to understand how different colorings of K_n relate to one another via transformations, which would allow us to partition the set of two colorings of K_n into a smaller number of classes and, in the absence of a full mathematical theory, to program the computer to search for the monochromatic complete subgraphs on the set of classes instead of the full set. Although still unsolved, intense activity in this area led to a number of generalizations of this problem and to the development of a huge branch of mathematics, the Ramsey theory (Graham et al. 1990). That theory has a number of very deep results that go well beyond graph theory, affecting set theory in



Pál Erdős (1913–1996)

Pál Erdős, who introduced the notion of random graphs, is probably the second most prolific mathematician of all times, having produced over 1500 publications with 507 coauthors.

the form of partition calculus, combinatorics, ergodic theory, logic, analysis, algebra, geometry and computer science.

Ultimately, the Party Problem suggests that, if we partition a set into a fixed number of classes, order must emerge for large enough sets. This principle is also illustrated by van der Waerden's theorem (Bollobás 1998), which states that, for a given k and p , if we partition the first w integers into k classes, we will always find a class that contains an arithmetic progression with p terms for large enough w . Problems like the Party Problem lead to a simple conclusion: In order to understand properties of graphs, one has to think in terms of ensembles of graphs that share a certain property, Q .

A Revolutionary Idea

The Hungarian mathematician Pál Erdős was one of the main pioneers of the ensemble approach. His complete disregard for the notion of possession

and ownership and his habit of living out of a suitcase and visiting one mathematician friend after the next were symptoms, perhaps, of his total dedication to mathematics. Erdős is considered by many to be the second most productive mathematician of all times, after Euler. Possibly the greatest contribution of Erdős is his introduction of the probabilistic method in discrete mathematics. For graph theory, this means that, instead of asking for detailed properties of all graphs in an ensemble, we are asking for average properties, or the probability that a graph from an ensemble has the property Q . The probabilistic method was definitely not new when Erdős introduced it to discrete mathematics: By the end of the 19th century, Boltzmann, Gibbs, and others had laid down the foundations of equilibrium statistical mechanics, which is based on applying the probabilistic method to ensembles of microstates and characterizing macroscopic properties of the system by the properties of the "typical" microstates. This natural connection between statistical mechanics and graph theory is currently being exploited by some research groups worldwide, including the Statistical Physics of Infrastructure Networks team at Los Alamos. Besides the combinatorial explosion in the number of possible graphs (or states), there is a second strong reason that calls for the use of the probabilistic method: incomplete information. Real-world networks, as we will see from the following sections, are in many cases very dynamic, with new edges and nodes appearing and old ones disappearing as a result of stochastic processes. In addition, in some cases, it is hard, or even impossible, to identify precisely the graph structure at a given moment. Again, a good example is supplied by a problem related to social networks, namely, the Gossip Problem:

Suppose that person A in a set of N people has a very interesting piece of information or gossip. On average, how many acquaintances must every person in N have such that the gossip becomes known to all?

Since we do not know who knows whom, we determine the answer by considering all graphs of N people with the nodes representing the individuals and the edges representing the acquaintanceships, or social links, for transmitting gossip. In other words, if persons A and B are linked and one of them knows the gossip, we can assume the other knows it too. The answer to the Gossip Problem must be probabilistic in nature: It is the class of graphs having nodes with a certain average number of links and characterized by the property that everyone knows the gossip in the end. Erdős and Rényi came up with a rather surprising solution: *Once a node has on average one link, the gossip becomes known to all!* In the jargon of social scientists, the set of people represented by that graph forms a society. The class of graphs that Erdős and Rényi introduced and that helped give the answer is called random graphs, a subject with a huge mathematical literature. For a review, see the book by Bollobás (2001).

The Binomial Random Graph.

Because we need it for later discussion, we introduce the binomial random graph $G(N,p)$ and present some of its properties. $G(N,p)$ is a class of graphs with N vertices, whose edges are drawn at random and independently, according to a uniform distribution with probability p . Therefore, the average number of links incident on a node is $\lambda = p(N-1)$, or for large graphs, it is approximately $\lambda = pN$. Thus, according to the answer for the Gossip Problem, when $\lambda = 1$, or the probability for a node to have an edge is $p = p_c = 1/N$, a giant cluster, or giant component, emerges that con-

tains most of the nodes, and the probability for a node not to belong to this cluster decreases exponentially fast for $p > p_c$. Physicists call this phenomenon percolation. Passing through p_c (by the process of increasing the average number of incident edges), the network suffers a drastic change, which is called a phase transition in the language of physics.

We now introduce one of the most important characteristics of random graphs, namely, their degree distribution. The degree of a node x is the number $k(x)$ of incident edges on that node. The degree distribution of the binomial random graph $G(N,p)$ is the probability that the number of nodes X_k with degree k is y . In a $G(N,p)$, the probability of a node being connected to k specific other nodes and not connected to the rest of $N-1-k$ nodes is $p^k(1-p)^{N-1-k}$. Because the number of ways to connect those k nodes is equal to the binomial coefficient

$$\binom{N-1}{k},$$

the probability of a node having exactly k incident edges in $G(N,p)$ becomes

$$P(k) = \binom{N-1}{k} p^k (1-p)^{N-1-k}. \quad (1)$$

Note that, as edges are drawn incident to a node, that node will influence the number of edges around the other nodes, and thus, in principle, the distribution of X_k will not be exactly the same as if all the nodes were independent, and the calculation of the exact form of the degree distribution becomes a hard task. It was Bollobás (2001) who showed that, for large enough N , the nodes can be treated as if they were independent, and thus, with good approximation, the degree distribution of $G(N,p)$ is described by

the binomial distribution in Equation (1). In the limit of $N \rightarrow \infty$ and $p \rightarrow 0$ such that $\lambda = pN = \text{constant}$, the binomial goes into the Poisson distribution:

$$P(k) \approx e^{-\lambda} \frac{\lambda^k}{k!}. \quad (2)$$

Figure 4 shows a comparison between the formula in (2) and the measured degree distribution for a binomial graph of $N = 20,000$ nodes and a link probability $p = 20/N = 0.001$. It shows that, indeed, the approximation is good. The Poisson distribution $P(k)$ has a “bell curve” shape, with a peak at $\lambda = pN$, and fast decaying tails. The degree of a node characterizes how a node “sees” its immediate neighborhood in the network. According to the formula in (2), if we keep $\lambda = pN$ a constant, while increasing the size of the network, the distribution of edges in the immediate neighborhood of a node becomes independent of N for large N . However, keeping $\lambda = pN$ a constant, means scaling the link probability p with $N-1$. The average node degree is

$$\langle k \rangle = \sum k P(k) = \lambda,$$

and the standard deviation of the distribution around the average is

$$\sigma = \sqrt{\lambda}.$$

This result shows that the binomial graph has a characteristic scale defined by λ .

Real-World Networks

The latest revolution in networks science happened toward the end of the 1990s, when powerful computers made it possible to gather and analyze data for systems containing a large

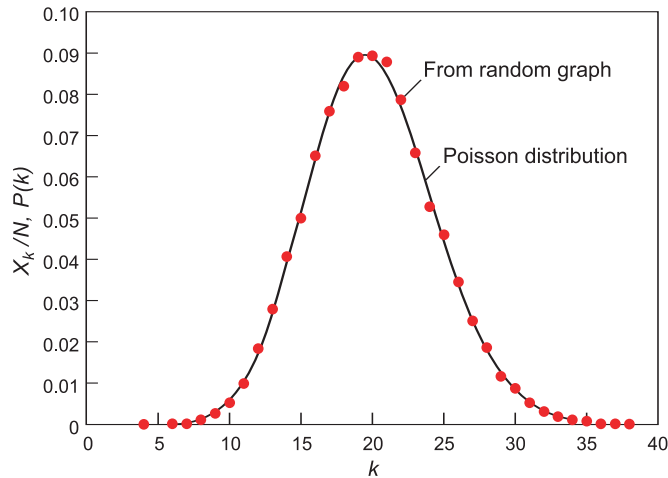


Figure 4. The Degree Distribution of the Binomial Random Graph
The red circles show the degree distribution, or X_k/N , the fraction of nodes with k links vs k , for a single instance of the binomial random graph $G(N,p)$ with the number of nodes $N = 2 \times 10^4$ and the probability for a link to exist between two nodes being $p = 10^{-3}$. The continuous black line is a plot of the Poisson distribution $P(k)$ in formula (2) with $\lambda = pN = 20$. Note the similarity between the two distributions.

number of components: from the World Wide Web and the Internet, phone call networks, networks of movie actors, large-company boards of directors, scientific collaboration networks, language networks, crime webs, epidemic networks, and the sex web to biological networks such as the metabolic network, protein interaction networks, cell-signaling, and food webs. The first important observation is that most of these networks are very different from the random graphs of Erdős and Rényi. In hindsight, this departure is not unexpected: In the random graphs of Erdős and Rényi, the edges are assumed to exist completely independently from each other, whereas in real-world networks, the existence of edges is typically conditioned by nonindependent processes, or constraints, such as spatial embedding and interaction range dependency. The real surprise is that, in spite of their diversity, real-world networks can be classified into a small number of different classes of graphs, each characterized by certain structural properties of the interaction topology

in these systems. The most useful properties for this purpose are degree distributions, clustering, assortativity, and shortest paths.

Instead of listing the classes of these networks and enumerating their properties, we will discuss one ubiquitous class, the so-called scale-free networks, originally introduced by Albert-László Barabási. These networks have power-law degree distributions (see Figure 5), as opposed to the Gaussian or Poisson degree distributions of random graphs (for example, Figure 4). These real-world, scale-free networks include the network of movie actors, scientific collaboration networks, the sex web, the metabolic network in the cell (on all three levels of life—archaea, bacteria, and eukaryotes), the protein interactions network, the language network defined by synonyms (in which case, the nodes are the words, and the edges connect the synonyms), and virtually all large-scale information networks: the Internet (router and also autonomous domain level), the World Wide Web, some e-mail networks and

phone call networks. Why do these real-world networks have similar degree distributions? Is there a universal mechanism that generates these structures? The first crucial observation is that, in most cases, these structures result from dynamic processes with a strong stochastic component, just like the random graph model of Erdős and Rényi. However, to deviate from the random graph model, the network evolution process must include stochastic dependency and bias. The question is then, “What stochastic processes will generate scale-free networks?”

Most current models generating scale-free networks identify a mechanism for network growth and evolution. Among the notable ones are the preferential attachment model of Albert and Barabási (2002), in which a newly arriving node connects to a node in the existing network with probability proportional to the current degree of the node in the network; the fitness-based network growth model of Caldarelli et al. (2002); the Chung-Lu model of power-law random graphs (2002); the model of the World Wide Web by Menczer (2002); the initial attractiveness model of Dorogovtsev et al. (2000); and others. Although these models produce graphs that have power-law degree distributions, they either have been built for a specific type of network (for example, the model by Menczer) or are mathematical abstractions in which the stochastic network-growth process has little to do with the actual, often quite complicated, evolution mechanism of the real-world network. The stochastic dynamics for the appearance and disappearance of Internet routers, which has many unknown factors, is another example. Most real-world networks are also strongly coupled to other networks or other large-scale complex systems, and thus, in order to identify the network evolution mechanism, one can-

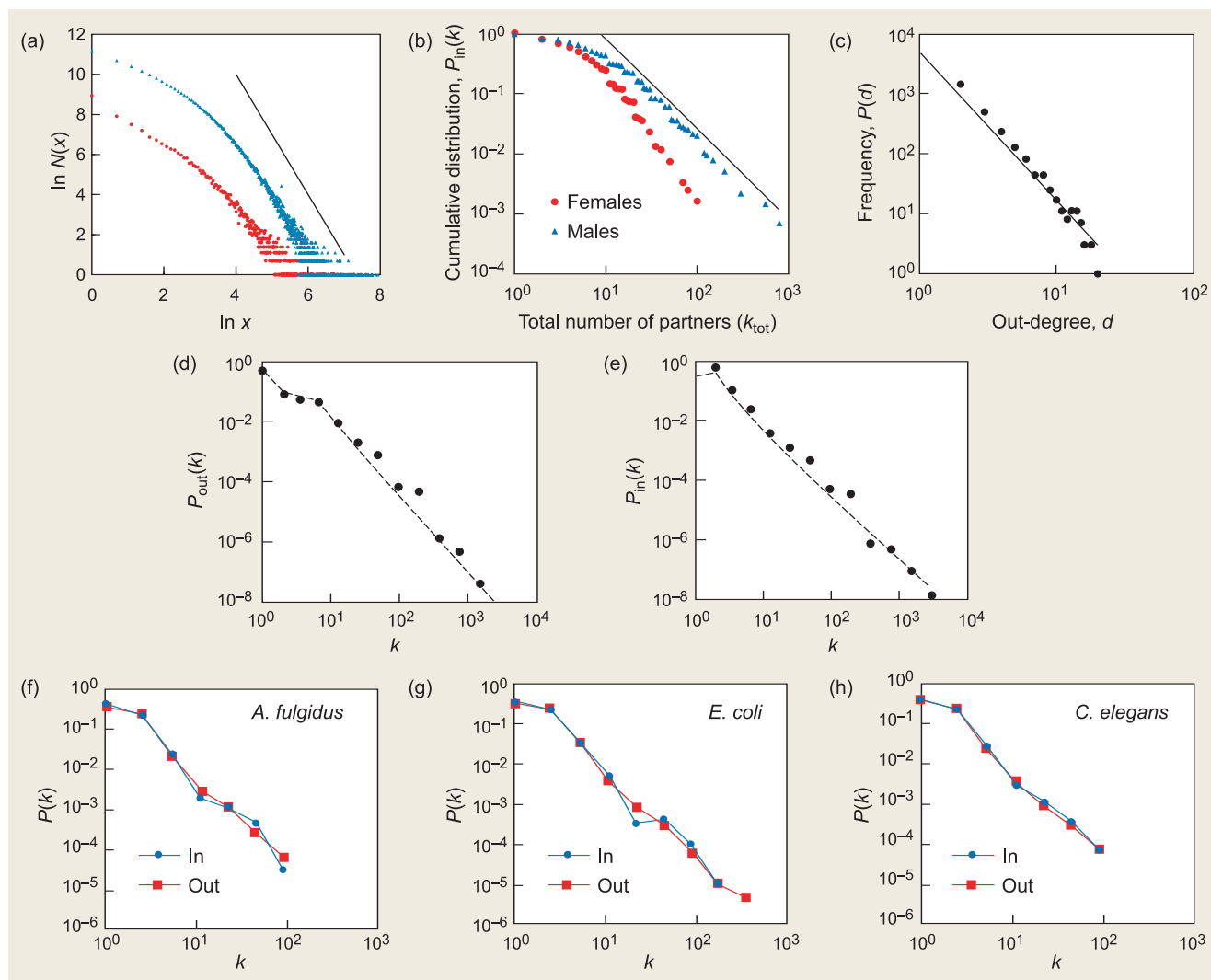


Figure 5. Degree Distribution of Various Scale-Free Networks

(a) Shown here is the cumulative degree distribution for citation networks (after Redner 1998); (b) the sex web (after Liljeros et al. 2001); (c) the Internet at the router level (after Faloutsos et al. 1999); (d) the in-link and (e) out-link degree distributions for the World Wide Web (after Albert et al. 1999); and (f) the metabolic networks for three species (after Jeong et al. 2000). [Plot (a) is courtesy of the *European Physical Journal B*. Plot (c) is courtesy of *Computer Communication Review*, ACM Publications 1999. Plots (b), (d), (e), (f), (g), and (h) are courtesy of *Nature*.]

not study these networks in isolation. To add to the complexity of the problem, the evolution of the network structure can depend on the dynamics or flow on the network. Most studies of complex networks have been static and structural as they try to identify their graph-theoretic properties. It has become clear, however, that, to solve even this problem, we must look at the full dynamics of the complex network, that is, at the flow on these

structures and the coupling of the flow to the structural evolution.

The Problem of Epidemics.

Before presenting some recent results that take into account the coupling between structure and dynamics, I will briefly mention an interesting and important real-world problem that is complex in the sense mentioned above. I am referring to epidemics, or disease propagation in living popula-

tions, a topic heavily studied at Los Alamos in the past decade. The usual, classic approach to epidemics imposes a number of assumptions that make analytic and numerical treatment relatively straightforward; however, at least in some cases, that approach may cause a departure from reality. One such assumption is uniform mixing, whereby the individuals of a population are assumed to come in contact with equal probability, independ-

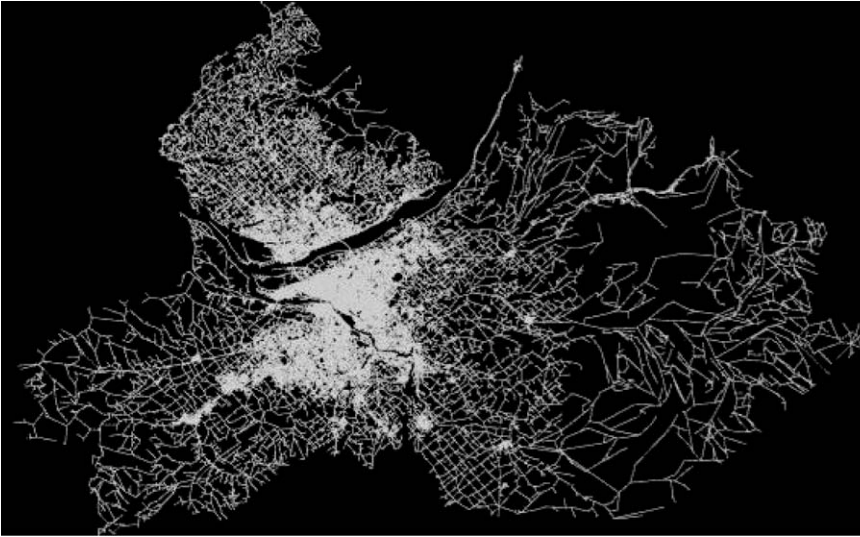


Figure 6. The Roadways of Portland, Oregon
The roadway network of Portland forms the substrate for a coupled complex dynamic network to simulate movements and disease transmission in this highly populated urban area. (This image is courtesy of the TRANSIMS Project at Los Alamos.)

ent of their locations. In order to relax this assumption, we observe that contact processes, such as disease transmission, are well localized in space and require that the two or more individuals be no farther apart than some typical distance characteristic of the disease transmission process. In heavily populated urban areas, disease is usually transmitted within such locations as buildings and mass transit areas (waiting areas and mass transit cars). Using census data and mobility diaries that specify the times of entrance and exit to and from a location for all locations that a specific person visited during the day, one obtains a graph that has the desired detailed resolution for contact patterns between people moving around in an urban area. This movement is largely constrained by the roadway network and the traffic on it. Since the roadway network is itself a complex network, the disease transmission problem is that of coupled complex dynamic networks (see Figure 6).

In this network, there are two types of nodes: people and locations, and an edge is drawn between a per-

son and a location if that person visited that location during the day. The edge has a weight associated with it, called “timestamp,” which is the union of time intervals during which the person was at that location. What can we learn, analyzing such a network, pertinent to disease outbreaks in a city? How can knowledge about this network be exploited to design effective vaccination and quarantine strategies? Conclusions for the specific case of Portland, Oregon, with approximately 1.6 million people and 181,000 locations can be found in Eubank et al. (2004).

Scale-Free Networks: Coincidence or Universality?

This section presents a different approach to understanding the emergence of the scale-free property for real-world networks. As mentioned previously, so far no one has found an obvious universal mechanism leading to power-law degree distributions for real-world networks. We actually suggest that, for a large class of networks

(to be specified below), there is no universal evolutionary mechanism. Instead, the network structure evolves according to a selection principle that promotes the global efficiency of transport and flow processing through these structures (Toroczkai and Bassler 2004). In other words, regardless of the specific evolutionary mechanism, that mechanism works within the constraints of the selection principle. And the operation of the selection principle on evolution often results in scale-free networks.

Most real-world networks (except those that are defined by artificial associations) serve as transport substrates for various entities such as information, energy, material, and forces. Some networks have evolved spontaneously (without global design), and it makes sense to enquire whether their dynamics obey a selection principle toward some kind of optimal or efficient behavior. Such a principle would be analogous to the one of natural selection that shapes both the biological networks at the cellular level and the food web.

Looking at the Internet, we note that, if a router receives too much traffic and causes constant congestion of the packets, engineers will fix the problem locally by bringing up more routers or modifying the routing algorithm. Similarly, in social networks, if an acquaintance does not satisfy our expectations about some set of social norms, that link will “naturally” be dropped from our own social network.

To explore this trend toward efficiency more formally, we first need to define a flow process on the network. Among the most ubiquitous flow processes in Nature are those generated by local variations, or gradients, of scalar quantities. Particle concentration, temperature, electric or gravitational potential, and pressure are just a few examples. The gradient-induced flow processes include granular flow, fluid flow, electric current, diffusion

processes, heat flow, and so on. Naturally, the same local-gradient mechanism will generate flows in complex networks. Two less obvious local-gradient examples are diffusive load balancing schemes used in distributed computation (Rabani et al. 1998), which are also employed in packet routing on the Internet, and the reinforcement learning mechanism in social networks with competitive dynamics (Anghel et al. 2004). In the first example, a computer (or a router) will ask its neighbors on the network for their current job load (or packet load), and the router will balance its load with the neighbor that has the minimum number of jobs to run (or packets to route). In this case, the scalar is the negative of the number of jobs, or packets, at nodes, and the flow will be along the direction of the gradient of this scalar in the node's network neighborhood. In the second example, a number of agents/players in the social network play an interactive competition game with limited information. At every step of the game, each agent has to decide whose advice to follow before taking an action (such as placing a bet), in its circle of acquaintances (network neighborhood). Typically, an agent will try to follow the advice of that neighbor who in the past proved to be the most successful in predicting the game. That neighbor is recognized by using a scoring mechanism, which is the simplest form of reinforcement learning: Every agent has a success score that changes in time, coupled to the game's evolution. An agent will follow the advice of that agent who has the highest score in its network neighborhood at that moment (Anghel et al. 2004). In this case, the scalar is the past success score of the agents, and an agent will act based on the information received along the link that is in the direction of the gradient of this scalar.

To construct a simple and general

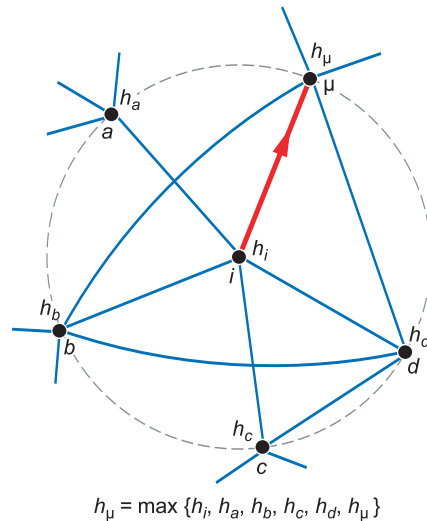


Figure 7. Definition of a Gradient Edge

The gradient edge is a directed link from node i to that neighbor on the substrate graph that has the largest value of the scalar in the neighborhood of i . If i has the largest value, then the gradient edge is a self-loop.

model of a transport process, we assume that there are N nodes and that the transport takes place on a fixed substrate network $S(V, E)$, where V is the node set and E is the edge set that describes the interconnections of the nodes. Associated with each node i , there is a scalar h_i that describes the “potential” of the node. Then a gradient network G can be constructed as the collection of directed links that point from each node to the nearest neighbor on the substrate network S that has the highest potential (see Figure 7). Thus, only one directed link points away from each node in G , and G consists of N directed links. Note that, if the potential of a node is higher than the potential of all its nearest neighbor nodes, the gradient link of that node is a loop that points back to itself (“self-loop”). In general, the potential for each point can evolve in time, and as a result, the gradient network G

will be time dependent. If we furthermore assume that all links have the same conductance, or transport properties, the gradient network will describe the instantaneous substructure carrying the maximum flow. Consequently, we can hope to use gradient networks as a tool to analyze the flow efficiency or susceptibility to jamming on the corresponding substrate networks.

Note that, if there are two or more nodes in the network neighborhood of a node i that share the maximum value, the gradient in i is called degenerate. If each neighborhood has only one maximum, it is called nondegenerate, and is easier to analyze. In the discussions below, we will restrict ourselves to the nondegenerate condition, which is easily realized if, for example, the scalars are continuous random variables. Since every node has exactly one gradient direction from it (even if it is a self-loop), G has exactly N nodes and N edges (and there is at least one self-loop, corresponding to $\max_i \{h_i\}$). A simple but very important property of nondegenerate gradient networks is that they form forests, that is, each gradient network is a collection of tree graphs containing no loops (except for self-loops). We can therefore hope to analyze network flow processes using the techniques of statistical mechanics that have been well developed for treelike structures.

Gradient Networks on Random vs Scale-free Networks. Let us first consider a gradient network for a random graph substrate S . In particular, we choose for S the binomial random graph, $G(N, p)$ consisting of N nodes, each pair of nodes being linked with probability p . We next assume that the scalar potentials of the various nodes are independent random variables identically distributed according to a distribution $\eta(h)$. The distribution of

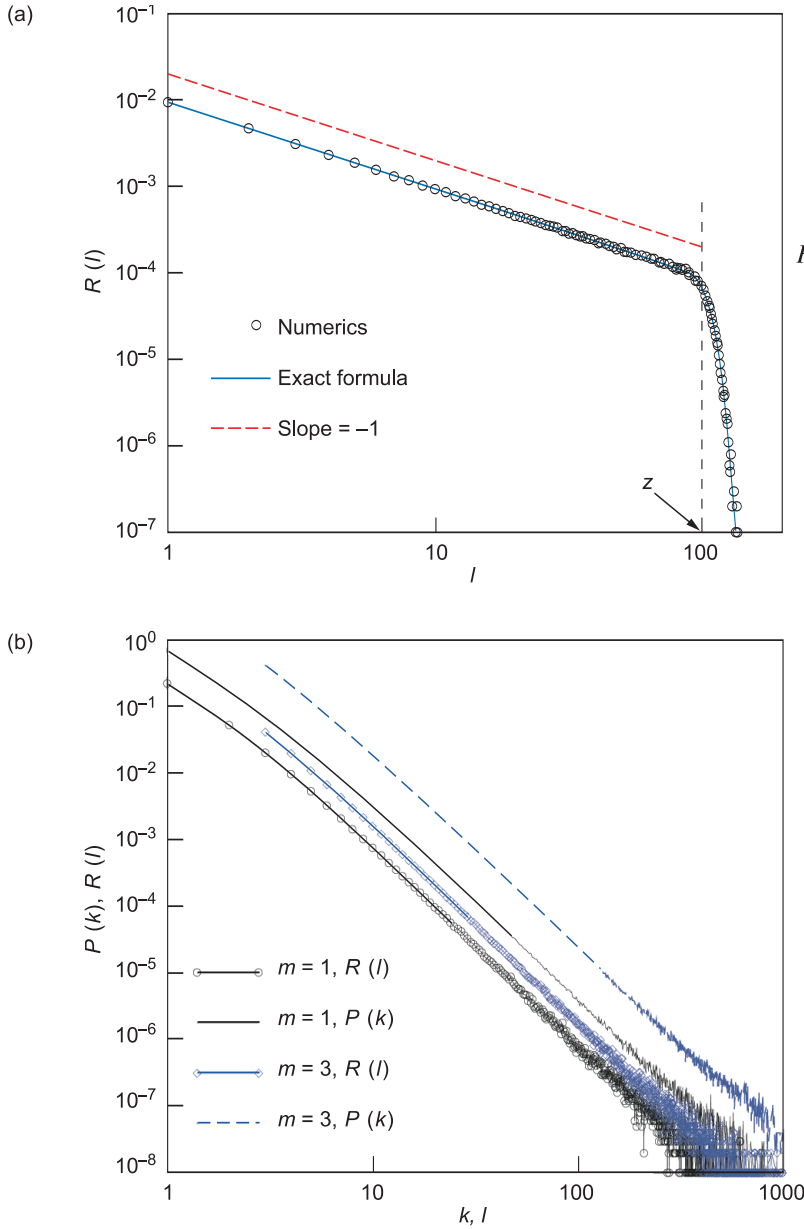


Figure 8. Gradient-Graph Degree Distributions for Random and Scale-Free Substrate Networks

(a) The in-degree distribution is shown for the substrate binomial random graph $G(N,p)$, where $N = 1000$, and $p = 0.1$ ($z = 100$). The numerical values are obtained after averaging over 10^4 sample runs. (b) The in-degree and degree $P(k)$ distributions are for the substrate Barabási-Albert scale-free graph with parameter m ($m = 1, 3$). In this case $N = 10^5$, and the average is performed over 10^3 samples.

the number of links l pointing to each node, the so-called in-degree distribution $R(l)$ of the gradient network G , can be exactly calculated, and it yields the following expression:

$$R(l) = \frac{1}{N} \sum_{n=0}^{N-1} \binom{N-1-n}{l} \times [1-p(1-p)^n]^{N-1-n-l} [p(1-p)^n]^l. \quad (3)$$

Thus, this in-degree distribution is independent of the particular form of the distribution for the scalar potentials $\eta(h)$. It is possible to show that in the limit $N \rightarrow \infty$ and $p \rightarrow 0$ such that $Np = \lambda = \text{constant} \gg 1$, the expression in Equation (3) becomes the power law $R(l) \approx 1/(\lambda l)$, with a finite-size cutoff at $l_c = z$; refer to Figure 8(a). Therefore, in this limit, gradient networks are scale-free graphs (up to their cutoff)! This power-law degree distribution for the gradient network is a rather surprising result because, in the same limit, the substrate graph S is a binomial random graph having a Poisson degree distribution with a well-defined average degree λ (setting the scale of the substrate graph), as well as rapidly decaying tails.¹

If, instead, the substrate network S is a scale-free graph, the gradient graph will still have a power-law degree distribution. Figure 8(b) compares degree distributions $P(k)$ for

¹A similar finding was reported by Lakhina et al. (2003), who repeated on binomial random graphs the trace-route measurements used to sample the structure of the Internet. Lakhina and colleagues found that the spanning trees obtained in this way have a degree distribution that obeys the $1/k$ law. Later, Clauset and Moore (2003) have presented an analytical approach to derive the $1/k$ law. This approach suggests the possibility of mapping between graphs generated by trace-route sampling and gradient networks. Although it is not an exact mapping, a close connection can indeed be made by interpreting trace-route trees as suitably constructed gradient networks.

scale-free substrate networks, which we generated by the Barabási-Albert network with parameter m (minimum degree, see Albert and Barabási 2002) with the in-degree distributions for the corresponding gradient networks. One immediate conclusion is that the gradient network is the same type of structure as the substrate. In this case, it is a scale-free (power-law) graph with the same exponent.

Flow Properties on Random vs Scale-Free Networks. Using the properties of gradient networks, we can define a transport characteristic related to congestion or jamming in the substrate network. In particular, we compare the average number of nodes with in-links with the average number of nodes with out-links. If $N_l^{(in)}$ denotes the number of nodes with l in-links, the total number of nodes receiving gradient flow will be

$$N_{\text{receive}} = \sum_{l \geq 1} N_l^{(in)} .$$

The total number of gradient out-links is simply $N_{\text{send}} = N$ because every node has exactly one out-link. Naturally, the ratio $N_{\text{receive}}/N_{\text{send}}$ will be related to the instantaneous global congestion in the network. The smaller the number of nodes receiving the flow (given the same number of senders), the more congestion is in the substrate network at that instant. If the flow received by a node requires a nonzero processing time (such as routing of a packet by the router), a small ratio of $N_{\text{receive}}/N_{\text{send}}$ translates into large delay times and thus inefficient flow processing. Let us define the congestion (or jamming) factor as follows:

$$J = 1 - \left\langle \left\langle \frac{N_{\text{receive}}}{N_{\text{send}}} \right\rangle_n \right\rangle_h = R(0) , \quad (4)$$

where $\langle \rangle_n$ means averaging over the disorder in the network structure and $\langle \rangle_h$ means averaging over the randomness in the scalar field. The value of J is always between 0 and 1, with $J = 1$ corresponding to maximal congestion and $J = 0$ corresponding to no congestion. Note that J is a congestion pressure characteristic generated by gradients rather than an actual throughput characteristic. For a binomial random substrate network $G(N,p)$, we use Equations (3) and (4) to obtain the corresponding jamming factor:

$$J^R(N,p) = \frac{1}{N} \sum_{n=1}^{N-1} [1-p(1-p)^n]^{N-1-n} . \quad (5)$$

In the scaling limit $N \rightarrow \infty$ and $p = \text{constant}$, the jamming factor assumes the asymptotic behavior

$$J^R(N,p) = 1 - \frac{\ln N}{N \ln \left(\frac{1}{1-p} \right)} \times \left[1 + O \left(\frac{1}{N} \right) \right] \rightarrow 1 .$$

That is, the random graph becomes maximally congested. It is easy to show that, in the other limit, when $z = Np \gg 1$ is kept constant while $N \rightarrow \infty$,

$$J^R(z) \approx 1 - \frac{\ln z}{z} - \dots \rightarrow 1 .$$

Once again, the random graph asymptotically becomes maximally congested, or jammed.

For scale-free networks, however, the conclusion about jamming is entirely different. We find that the jamming coefficient J becomes independent of N , and it is always a

constant less than unity for large networks. In other words, scale-free networks are not prone to maximal congestion. (This is true for all power-law networks for which the average degree does not grow with N .) Figure 9 shows the congestion factors as a function of network size for random and scale-free substrate networks. Many real-world networks evolve more or less spontaneously (for example, the Internet or the World Wide Web), and they can reach sizes of about 10^8 nodes. At such large N , the scaling limit studied above applies, and random networks have maximal congestion. Thus, such substrates are very inefficient for flow processing. Scale-free networks, on the other hand, have congestion that stays bounded away from unity as the number of nodes grows very large, and they are therefore much more efficient substrates for transport and flow processing. Thus, it appears that the scale-free property of many real-world networks is not accidental. Topology may develop quite naturally from a selection rule that tends to maximize the global efficiency of the flow along the network.

Small-World Magic: Synchronized Computing Networks

We have seen that many real-world interactions are mediated across complex network topologies and that the structure and dynamics of those complex networks are becoming better understood. It is therefore natural to wonder whether network concepts can be put to practical use. For example, can those concepts help us design systems that exhibit certain desired properties? In this last section of the article, I will show how complex network concepts were used to solve a problem in distributed, or parallel, computation.

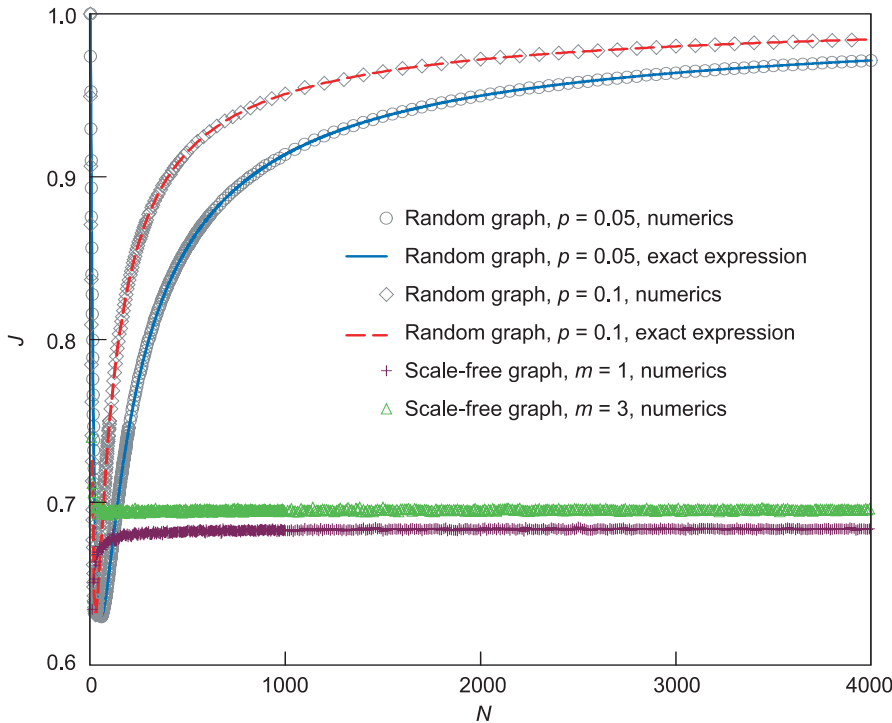


Figure 9. Congestion Factors for Random and Scale-Free Substrate Networks

Congestion factors are shown as a function of size for random graphs and scale-free networks. For random binomial graph substrates, the jamming coefficient tends to unity with increasing network size, indicating that these networks will become extremely congested in this limit. For scale-free substrates, however, the congestion factor becomes independent on the network size, and thus arbitrarily large networks can be considered without increasing their congestion level.

We consider the class of systems made of a large number of interacting elements or individuals, each having a finite number of attributes, or local state variables, that can assume a countable number (typically finite) of values. The dynamics of the local state variables are discrete events occurring in continuous time, and the interactions between individuals, or elements, have a finite range. There are many examples of such systems: magnetic systems, epidemics, some financial markets, wireless communications, queuing systems, and so on. Virtually all agent-based systems can be considered to belong to this class of discrete-event complex systems. Often, the dynamics of such systems is inherently stochastic and asynchro-

nous. Simulating the systems is nontrivial, and in most cases, the complexity of the problem requires the use of distributed computer architectures. These problems define the field of parallel discrete-event simulations (PDES).

Conceptually, the computational task is divided among N processing elements (PEs), each of which evolves the dynamics of the allocated piece of the system. Because of the interactions among the individual elements of the real system (spins, atoms, packets, calls, and so on), the PEs must coordinate with a subset of other PEs during the simulation.

At present, large parallel computers for performing PDES have thousands of nodes and soon will have

tens of thousands: the Nippon Electric Company's 5120-node Earth simulator producing 35.86 teraflops, the 8192-node Q-machine at Los Alamos with 13.88 teraflops, Virginia Tech's X machine, which is a 2200-node apple G5 cluster with 10.28 teraflops, and so on. IBM is currently building the Blue Gene/L parallel computer with 360 teraflops and 65,000 nodes. Blue Gene/P, the next-generation computer, is expected to surpass the petaflop barrier in 2006.

The design of efficient, scalable update schemes for performing PDES on these large parallel computers is a rather challenging problem because the simulation scheme itself becomes a complex system whose properties are hard to deduce using classical methods of algorithm analysis. Korniss et al. (2003) introduced a less conventional approach to analyzing the efficiency and scalability of parallel discrete-event simulation schemes. The authors constructed an exact mapping between the parallel computational process itself and a nonequilibrium surface growth model. As a result, questions about efficiency and scalability can be mapped into certain topological properties of this nonequilibrium surface. Then, using methods from statistical mechanics, we can solve the scalability problem of the computation PDES schemes. We now briefly sketch the scalability problem and its solution.

In order to simulate the dynamics of the underlying system, the PDES scheme must track the physical-time variable of the complex system. In case of asynchronous dynamics on distributed architectures, each PE generates its own physical (also called virtual) time τ , which is the physical time variable of the particular computational domain handled by that PE. Because of the varying complexity of the computation at different PEs, at a given wall-clock instant, the simulated, or virtual, times of the PEs can

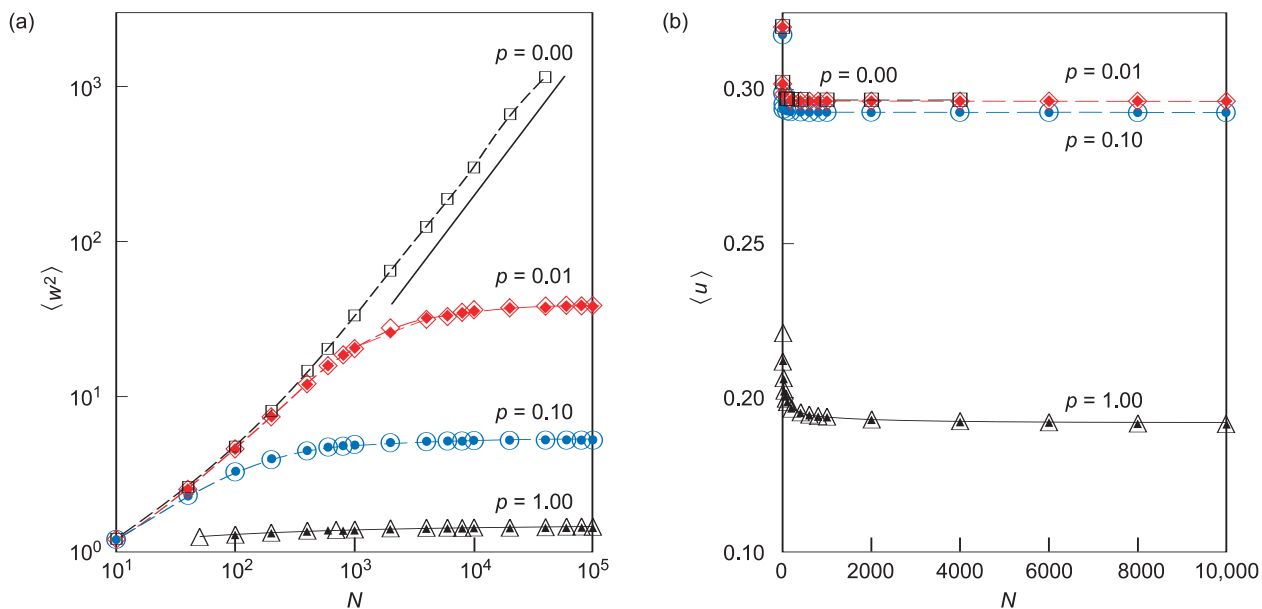


Figure 10. A Fully Scalable Small-World PDES Scheme

The small-world PDES scheme is fully scalable. By introducing more shortcuts into the communication network (increasing ρ), the algorithm becomes measurement scalable (a), and it stays computationally scalable (b).

differ, a phenomenon called “time horizon roughening.” Let us denote the virtual time at PE_{*i*} measured at wall-clock time t by $\tau_i(t)$. The set of virtual times $\{\tau_i(t)\}_{i=1}^N$ forms the virtual time horizon of the PDES scheme after t parallel updates. In conservative PDES schemes, a PE will perform its next update only if it can obtain the correct information from its neighbors to evolve the local configuration (local state) of the underlying physical system it simulates without violating causality. Otherwise, it idles. Specifically, the PE_{*i*} can only update (become “active”) at wall-clock instant t if

$$\tau_i(t) \leq \min_{j \in \langle i \rangle} \{\tau_j(t)\}. \quad (6)$$

That is, the PE’s virtual time is a local minimum among the virtual times of its neighboring PEs (specified as the set $\langle i \rangle$). Once the PE at site i can update, it will advance its local simu-

lated time to the new value $\tau_i(t+1)$, and the process is repeated for all active sites, generating the dynamics of the virtual time horizon $\{\tau_i(t)\}_{i=1}^N$. The average of the time horizon after t parallel steps is obviously

$$\bar{\tau}(t) = \frac{1}{N} \sum_{j=1}^N \tau_j(t)$$

Thus, the rate of progress of the time horizon average, or the average utilization of the PEs $\langle u(t) \rangle = \langle \bar{\tau}(t+1) \rangle - \langle \bar{\tau}(t) \rangle$ is proportional to the number of nonidling, or active, PEs. The average $\langle \cdot \rangle$ is taken over the stochastic event dynamics. The PDES scheme is computationally scalable if there is a constant $c > 0$, such that

$$\langle u(\infty) \rangle = \lim_{t, N \rightarrow \infty} \langle u(t) \rangle > c. \quad (7)$$

That is, the average rate of progress of the time horizon does not vanish even

after very long times, as the simulated system size and, therefore, the number of PEs are taken to infinity.

We solved this computational scalability problem by drawing an analogy with the statistical mechanics of non-equilibrium surface-growth processes. Thin films are grown on solid substrates by deposition of atoms or molecules from surrounding vapors. Because the vapors are fairly hot, the atoms reaching the solid surface follow a stochastic path until they are incorporated into the surface, typically in an irregular fashion. The resulting thin film has mounds and valleys that can be described by the fluctuations of the local height variable $h(x,t)$ of the film measured from the surface of the substrate. Using an approach based on the Langevin equation, physicists have developed extended theoretical machinery to describe the statistics of the fluctuations of the variable $h(x,t)$. The simulated time variable $\tau_i(t)$ in the computational scalability problem

behaves much like the surface height variable $h(x,t)$ in that $\tau_i(t)$ evolves according to the stochastic update dynamics of the PDES scheme with the index i of the PE corresponding to x in the height variable. In many large complex systems, the dynamics of the stochastic events can be characterized by a Poisson distributed stream. This means that, when simulating such systems, the updates at individual PEs correspond to adding height increments that follow a Poisson distribution. Using statistical mechanics methods to analyze the resulting surface-growth model, one can show that the fluctuations of the virtual time horizon in the continuum limit can be described by the so-called Kardar-Parisi-Zhang (KPZ) equation of surface growth:

$$\frac{\partial \hat{\tau}}{\partial t} = \frac{\partial^2 \hat{\tau}}{\partial x^2} - \lambda \left(\frac{\partial \hat{\tau}}{\partial x} \right)^2 + \eta(x,t), \quad (8)$$

where $\hat{\tau}$ is a coarse-grained form of the virtual-time variable and η is a white noise term.² We then use the KPZ equation to verify that the utilization of the PEs satisfies Equation (7). The existence of a constant $c > 0$, as in (7), is the result of the slope-slope correlations of the surface being short ranged and not scaling with N . Our numerical evaluation of this constant yields $c = 0.2461 \pm (7 \times 10^{-6})$, which shows that the basic conservative PDES scheme is indeed computationally scalable.

There is, however, a fundamental problem with the basic PDES update scheme. The KPZ equation for the time horizon fluctuations predicts that the average spread of those fluctuations, $w^2(N,t)$, diverges with an increasing number of processing

elements N in the long time limit ($t \rightarrow \infty$). Therefore, if we try to measure a global property of simulated system at a given simulated time τ and wait until all processors have simulated their local state corresponding to that time, the waiting period in wall-clock time would diverge with the number of processing elements! In other words, even though a parallel computer with infinitely many processing elements can simulate the dynamics of an infinitely large system at nonzero speed (computational scalability), the basic PDES scheme could not produce a single measurement of the global state of the system! The basic conservative scheme is computationally scalable but measurement nonscalable.

How can we surmount this problem? Can the PDES scheme be modified such that the new update scheme is also measurement scalable? The answer is affirmative, and the key to the solution is the notion of the small-world property of complex networks (Korniss et al 2003).

In order to decorrelate the fluctuations in the time horizon, we modify the update topology in the following way: for every node i , we assign a randomly chosen communication link, $r(i)$. According to its definition, the resulting communication topology (a regular lattice plus random links) forms a small-world network. When a node is allowed to update—its virtual time satisfies the condition in (6)—it will make, with probability p , an extra check for the condition $\tau_i(t) \leq \tau_{r(i)}(t)$ and update if that condition is satisfied. With probability $1 - p$, it will make this extra check and thus behave as the basic PDES scheme. Here p has the role of a tuning parameter: For $p = 0$, we have the basic PDES scheme, whereas $p = 1$ corresponds to the fully scalable small-world PDES scheme. Note that these extra checks do not affect the correctness of the simulation, and

causality is preserved in just the same way. These checks only synchronize the PEs. Using the same coarse-graining methods as for the basic PDES scheme, we now find that the time horizon fluctuations are described by

$$\frac{\partial \hat{\tau}}{\partial t} = -\gamma(p)\hat{\tau} + \frac{\partial^2 \hat{\tau}}{\partial x^2} - \lambda \left(\frac{\partial \hat{\tau}}{\partial x} \right)^2 + \eta(x,t), \quad (9)$$

with $\gamma(0) = 0$ and $\gamma(p) > 0$ for $0 < p \leq 1$. This equation differs from Equation (8) in the strong damping term, $-\gamma(p)\hat{\tau}$, which is ultimately responsible for the nondivergence of the average spread, and thus the new update scheme is measurement scalable as shown in Figure 10.

Concluding Remarks

The list of problems, challenges, and applications that I presented above is rather biased toward my particular research interests, and it is not, by far, exhaustive of this area. My goal was to give the reader a feeling for the type of complexity one encounters when dealing with networks. I also wanted to show that this is a novel area with many interesting and potentially powerful applications awaiting discovery. ■

Further Reading

- Albert, R., and A.-L. Barabási. 2002. Statistical Mechanics of Complex Networks. *Rev. Mod. Phys.* **74**: 47.
- Albert, R., H. Jeong, and A.-L. Barabási. 1999. Diameter of the World-Wide Web. *Nature* **401**: 130.
- Anghel, M., Z. Toroczkai, K. E. Bassler, and G. Korniss. 2004. Competition-Driven Network Dynamics: Emergence of a Scale-Free Leadership Structure and Collective Efficiency. *Phys. Rev. Lett.* **92**: 058701.

² The deterministic part of this equation is easily related to the well-known Burgers equation through the slope variables $\phi = \partial \hat{\tau} / \partial x$.

- Bollobás, B. 1998. *Modern Graph Theory: Graduate Texts in Mathematics*. Vol. 184. Edited by F. W. Gehring, and S. Axler. New York: Springer-Verlag.
- . 2001. *Random Graphs*. Second Edition. Cambridge; New York: Cambridge University Press.
- Caldarelli, G., A. Capocci, P. De Los Rios, and M. A. Muñoz. 2002. Scale-Free Networks from Varying Vertex Intrinsic Fitness. *Phys. Rev. Lett.* **89**: 258702.
- Chung, F., and L. Lu. 2002. Connected Components in Random Graphs with Given Expected Degree Sequences. *Ann. Comb.* **6**: 125.
- Clauset, A. and C. Moore. 2004. Traceroute Sampling Makes Random Graphs Appear to Have Power Law Degree Distributions. [Online]: <http://arxiv.org/abscondmat/0312674>
- Dorogovtsev, S. N., J. F. F. Mendes, and A. N. Samukhin. 2000. Structure of Growing Networks with Preferential Linking. *Phys. Rev. Lett.* **85** (21): 4633.
- Eubank, S., H. Guclu, V. S. A. Kumar, M. V. Maratho, A. Srinivasan, Z. Toroczkai, and N. Wang. 2004. Modelling Disease Outbreaks in Realistic Urban Social Networks. *Nature* **429**: 180.
- Faloutsos, M., P. Faloutsos, and C. Faloutsos. 1999. On Power-Law Relationships of the Internet Topology. *Comput. Commun. Rev.* **29** (4): 251.
- Graham, R. L., B. L. Rothschild, and J. H. Spencer. 1990. *Ramsey Theory*. Second Edition. New York: John Wiley and Sons, Inc.
- Jeong, H., B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. 2000. The Large-Scale Organization of Metabolic Networks. *Nature* **407**: 651.
- Kilian, J., Y. Rabani, A. Sinclair, and R. Wanka. 1998. Local Divergence of Markov Chains and the Analysis of Iterative Load Balancing Schemes. In *Proceedings of the 39th Annual Symposium on the Foundations of Computer Science (FOCS)*. p. 694. Los Alamitos, CA: IEEE Computer Society.
- Korniss, G., M. A. Novotny, H. Guclu, Z. Toroczkai, and P. A. Rikvold. 2003. Suppressing Roughness of Virtual Times in Parallel Discrete-Event Simulations. *Science* **299** (5607): 677.
- Lakhina, A., J. W. Byers, M. Crovella, and P. Xie. 2003. Sampling Biases in IP Topology Measurements. In *Proceedings of the Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*. p. 332. New York: IEEE.
- Liljeros, F., C. R. Edling, L. A. N. Amaral, H. E. Stanley, and Y. Åberg. 2001. The Web of Human Sexual Contacts. *Nature* **411**: 907.
- Lloyd, S. 2000. Ultimate Physical Limits to Computation. *Nature* **406**: 1047.
- Menczer, F. 2002. Growing and Navigating the Small World Web by Local Content. *Proc. Natl. Acad. Sci. U.S.A.* **99** (22): 14014.
- Redner, S. 1998. How Popular Is Your Paper? An Empirical Study of the Citation Distribution. *Eur. Phys. J. B*, **4** (2): 132.
- Solomonoff, R., and A. Rapoport. 1951. Connectivity of Random Nets. *Bull. Math. Biophys.* **13**: 107.
- Toroczkai, Z., and K. E. Bassler. 2004. Jamming is Limited in Scale-Free Systems. *Nature* **428**: 716.

*For further information, contact
Zoltán Toroczkai (505) 667-3218
(toro@lanl.gov).*