Why We Don't Know How To Simulate The Internet

Vern Paxson and Sally Floyd*
Network Research Group
Lawrence Berkeley National Laboratory
University of California, Berkeley 94720, U.S.A.
{vern,floyd}@ee.lbl.gov

LBNL-41196

December 1997

Abstract

Simulating how the global Internet data network behaves is an immensely challenging undertaking because of the network's great heterogeneity and rapid change. The heterogeneity ranges from the individual links that carry the network's traffic, to the protocols that interoperate over the links, to the "mix" of different applications used at a site and the levels of congestion (load) seen on different links. We discuss two key strategies for developing meaningful simulations in the face of these difficulties: searching for invariants and judiciously exploring the simulation parameter space. We finish with a look at a collaborative effort to build a common simulation environment for conducting Internet studies.

1 Introduction

The Internet is a global data network connecting millions of computers, and is rapidly growing. In this paper, we discuss why simulating its behavior is an immensely challenging undertaking. First, for the reader unacquainted with how the Internet works, we give a brief overview of its operation.

Whenever Internet computers wish to communicate with one another, they divide the data they wish to exchange into a sequence of "packets" that they inject into the network. The Internet's infrastructure consists of a series of "routers" interconnected by "links." The routers examine each packet they receive in order to determine the next "hop" (either another router or the destination computer) to which they should forward the packet so that it will ultimately reach its destination. Sometimes routers receive more packets than they can immediately forward, in which case they momentarily *queue* the data in "buffers," increasing the delay of the packets through the network; or sometimes they must *drop* incoming packets, not forwarding them at all (not a rare event).

The specifics of how to format individual packets for transmission through the network form one of the Internet's underlying "protocols" (this fundamental one is called the Internet Protocol, or IP). Other protocols regulate other facets of Internet communication, such as how to divide streams of data into individual packets such that the original data can be delivered to the receiving computer intact, even if some of the individual packets are lost due to drops or damage (a form of "transport" protocol, which is built on top of IP). Still other "application" protocols are built on top of different transport protocols, providing network services such as email or access to the World Wide Web (WWW).

The design of the Internet continues to evolve, and many aspects of its behavior are poorly understood. Due to the network's complexity, simulation plays a vital role in attempting to characterize how different facets of the network behave, and how proposed changes might affect the network's different properties. Yet simulating different aspects of the Internet is exceedingly difficult. In this paper we endeavor to explain the underlying difficulties (§2–§5), which are rooted in the network's immense heterogeneity and the great degree to which it changes over time, and then discuss some strategies for accommodating these difficulties (§6), as well as taking a brief look at how one collaborative effort is attempting to advance the state-of-the-art (§7). We finish in §8 with a discussion of how simulation fits in with other forms of Internet research.

2 An Immense Moving Target

The Internet has several key properties that make it exceedingly hard to characterize, and thus to simulate. First, its great success has come in large part because the main function of the IP architecture is to unify diverse networking technologies and administrative domains. IP allows vastly

*This work was supported by the Director, Office of Energy Research, Office of Computational and Technology Research, Mathematical, Information, and Computational Sciences Division of the United States Department of Energy under Contract No. DE-AC03-76SF00098. This paper appears in the *Proceedings of the 1997 Winter Simulation Conference*, Atlanta, GA, 1997.

different networks administered by vastly different policies to seamlessly interoperate. However, the fact that IP masks these differences from a *user*'s perspective does not make them go away! IP buys uniform *connectivity* in the face of diversity, not uniform *behavior*.

A second key property is that the Internet is big. The most recent estimate is that it included more than 16 million computers in Jan. 1997 (Lottor, 1997). Its size brings with it two difficulties. The first is that the range of heterogeneity mentioned above is very large: if only a small fraction of the computers behave in an atypical fashion, the Internet still might include thousands of such computers, often too many to dismiss as negligible.

Size also brings with it the crucial problem of *scaling*: many networking protocols and mechanisms work fine for small networks of tens or hundreds of computers, or even perhaps "large" networks of 10's of 1,000's of computers, yet become impractical when the network is again three orders of magnitude larger (and likely to be five orders of magnitude larger within a decade). Large scale means that rare events *will* routinely occur in some part of the network, and, furthermore, that reliance on human intervention to maintain critical network properties such as stability becomes a recipe for disaster.

A third key property is that the Internet changes in *drastic* ways over time. For example, we mentioned above that in Jan. 1997, the network included 16 million computers. If we step back a year in time to Jan. 1996, however, then we find it included "only" 9 million computers. This observation then begs the question: how big will it be in another year? 3 years? 5 years? One might be tempted to dismiss its near-doubling during 1996 as surely a one-time phenomenon. However, this is not the case. For example, Paxson (1994a) discusses measurements showing sustained Internet traffic growth of 80%/year *going back to 1984*. Accordingly, we *cannot* assume that the network's current, fairly immense size indicates that its growth must surely begin to slow.

Unfortunately, growth over time is not the only way in which the Internet is a moving target. Even what we would assume must certainly be solid, unchanging statistical properties can change in a brief amount of time. For example, in Oct. 1992 the median size of an Internet FTP (file transfer) connection observed at LBNL was 4,500 bytes (Paxson, 1994b). The median is considered a highly *robust* statistic, one immune to outliers (unlike the mean, for example), and in this case was computed over 60,000 samples. Surely this statistic should give us some solid predictive power in forecasting future FTP connection characteristics! Yet only five months later, the same statistic computed over 80,000 samples yielded 2,100 bytes, less than half what was observed before. Thus, we must exercise great caution in assuming that observations made at a particular point in time tell us much about properties at other points in time.

For Internet engineering, however, the growth in size and change in connection characteristics in some sense pale when compared to another way in which the Internet is a moving target: it is subject to major changes in *how* it is used, with new applications sometimes virtually exploding on the scene and rapidly altering the lay of the land. For example, for a research site studied by Paxson (1994a), the Web was essentially unknown until late 1992 (and other traffic dominated the site). Then, a stunning pattern of growth set in: the site's Web traffic began to *double every six weeks*, and continued to do so for *two full years*. Clearly, any predictions of the shape of future traffic made before 1993 were hopelessly off the mark by 1994, when Web traffic wholly dominated the site's activities.

Furthermore, such explosive growth was not a one-time event associated with the paradigm-shift in Internet use introduced by the Web. For example, in Jan. 1992 the MBone—a "multicast backbone" used to transmit audio and video over the Internet—did not exist. Three years later, it made up 20% of all of the Internet data bytes at one research lab; 40% at another; and more than 50% at a third. It too, like the Web, had exploded. In this case, however, the explosion abated, and today MBone traffic is overshadowed by Web traffic. How this will look tomorrow, however, is anyone's guess.

In summary: the Internet's technical and administrative diversity, sustained growth over time, and immense variations over time in which applications are used and in what fashion, all present immense difficulties for attempts to simulate it with a goal of obtaining "general" results.

3 Heterogeneity Any Which Way You Look

Even if we fix our interest to a single point of time, the Internet remains immensely heterogeneous. In the previous section we discussed this problem in high-level terms; here, we discuss more specific areas in which ignoring heterogeneity can completely undermine the strength of simulation results.

3.1 Topology and link properties

A basic question for a network simulation is what topology to use for the network being simulated—the specifics of how the computers in the network are connected (directly or indirectly) with each other, and the properties of the links that foster the interconnection.

Unfortunately, the topology of the Internet is difficult to characterize. First, it is constantly changing. Second, the topology is engineered by a number of different entities, not all of whom are willing to provide topological information. Because there is no such thing as a "typical" Internet topology, simulations exploring protocols that are sensitive to topological structure can at best hope to characterize how the protocol performs in a range of topologies.

On the plus side, the research community has made significant advances in developing topology-generators for Internet simulations (Calvert, Doar and Zegura, 1997). Several of the topology generators can create networks with locality and hierarchy loosely based on the structure of the current Internet.

The next problem is that while the properties of the different types of links used in the network are generally known, they span a very large range. Some are slow modems, capable of moving only hundreds of bytes per second, while others are state-of-the-art fiber optic links with *bandwidths* a million times faster. Some are "point-to-point" links that directly connect two computers (this form of link is widely assumed in simulation studies); others are "broadcast" links that directly connect a large number of computers (these are quite common in practice).

Another type of link is that provided by connections to satellites. If a satellite is in geosynchronous orbit, then the *latency* up to and back down from the satellite will be on the order of 100's of milliseconds, much higher than for most land-based links. On the other hand, if the satellite is in low-earth orbit, the latency is quite a bit smaller, but *changes* with time as the satellite crosses the face of the earth.

Another facet of topology easy to overlook is that of *dynamic routing*. In the Internet, routes through the network can *change* on time scales of seconds to days (Paxson, 1996), and hence the topology is not fixed. If the route changes occur on fine enough time scales (per-packet changes are not unknown), then one must refine the notion of "topology" to include "multi-pathing." Multi-pathing immediately brings other complications: the latency, bandwidth and load of the different paths through the network might differ considerably.

Finally, routes are often *asymmetric*, with the route from computer A to computer B through the network differing in the hops it visits from the reverse route from B to A. Routing asymmetry can lead to asymmetry in path properties such as bandwidth (which can also arise from other mechanisms). An interesting facet of asymmetry is that it often only arises in large topologies: it provides a good example of how *scaling* can lead to unanticipated problems.

3.2 Protocol differences

Once all of these topology and link property headaches have been sorted out, the researcher conducting a simulation study must then tackle the specifics of the protocols used in the study. For some studies, simplified versions of the relevant Internet protocols may work fine. But for other studies that are sensitive to the details of the protocols (it can be hard to tell these from the former!), the researcher faces some hard choices. While conceptually the Internet uses a unified set of protocols, in reality each protocol has been implemented by many different communities, often with significantly different features (and of course bugs). For example, the widely used Transmission Control Protocol (TCP) has undergone major evolutionary changes. A study of eleven different TCP implementations found distinguishing differences among nearly all of them (Paxson, 1997), and major problems with several.

Consequently, researchers must decide which real-world features and peculiarities to include in their study, and which can be safely ignored. For some simulation scenarios, the choice between these is clear; for others, determining what can be ignored can present considerable difficulties.

After deciding which specific Internet protocols to use, they must then decide which *applications* to simulate using those protocols. Unfortunately, different applications have major differences in their characteristics; worse, these characteristics vary considerably from site to site, as does the "mix" of which applications are predominantly used at a site. Again, researchers are faced with hard decisions about how to keep their simulations tractable without oversimplifying their results to the point of uselessness.

4 Traffic Generation

For many Internet simulations, a basic problem is how to introduce different traffic sources into the simulation. The difficulty with synthesizing such traffic is that no solid, abstract description of Internet traffic exists. At best, some (but not all) of the salient characteristics of such traffic have been described in abstract terms, a point we return to in §6.1.

Trace-driven simulation might appear at first to provide a cure-all for the heterogeneity and "real-world warts and all" problems that undermine abstract descriptions of Internet traffic. If only one could collect enough diverse traces, one could in principle capture the full diversity. This vision fails for a basic, often unappreciated reason. One crucial property of much of the traffic in the Internet is that it uses *adaptive congestion control*. That is, each source transmitting data over the network inspects the progress of the data transfer so far, and if it detects signs that the network is under stress, it cuts the rate at which it sends data, in order to do its part in diminishing the stress (Jacobson, 1988). Consequently, the timing of a connection's packets as recorded in a trace intimately reflects the conditions in the network at the time the connection occurred. Furthermore, these conditions are *not* readily determined by inspecting the trace. Connections adapt to network congestion anywhere along the end-to-end path between the sender and the receiver. So a connection observed on a high-speed, unloaded link might still send its packets at a rate much lower than what the link could sustain, because somewhere else along the path insufficient resources are available for allowing the connection to proceed faster.

In this paper we will refer to this phenomenon as traffic *shaping*. Traffic shaping leads to a dangerous pitfall when simulating the Internet, namely the temptation to use trace-driven simulation to incorporate the diverse real-world effects seen in the network. The key point is that, due to rate adaptation, we cannot safely reuse a trace of a connection's packets in another context, because the connection would not have behaved the same way in the new context!

Traffic shaping does *not* mean that, from a simulation perspective, measuring traffic is fruitless. Instead, we must shift our thinking away from trace-driven *packet-level* simulation and instead to trace-driven *source-level* simulation. That is, for most applications, the volumes of data sent by the endpoints, and often the application-level pattern in which data is sent (request/reply patterns, for example), is not shaped by the network's current properties; only the lower-level specifics of exactly *which* packets are sent *when* are shaped. Thus, if we take care to use traffic traces to characterize *source behavior*, rather than packet-level behavior, we can then use the source-level descriptions in simulations to synthesize plausible traffic. See Danzig et al. (1992), Paxson (1994b), and Cunha, Bestavros and Crovella (1995).

An alternative approach to deriving source models from traffic traces is to characterize traffic sources in more abstract terms, such as using many data transfers of a fixed size or type. The Internet's pervasive heterogeneity raises the question: *which* set of abstractions should be used? Is the traffic of interest dominated by, for example, the aggregate of thousands of small connections (Web "mice"), or a few extremely large, one-way, rate-adapting bulk transfers ("elephants"), or long-running, high-volume video streams "multicasted" from one sender to multiple destinations, or bidirectional multimedia traffic generated by interactive gaming?

A final dimension that must be explored is: to what level should the traffic congest the network links? Virtually all degrees of congestion, including none at all, are observed with non-negligible probability in the Internet. More generally, variants on the above scenarios all occur in different situations frequently enough that they cannot be dismissed out of hand.

5 Today's Network Is Not Tomorrow's

A harder issue to address in a simulation study concerns how the Internet might evolve in the future. For example, all of the following might or might not happen within the next year or two:

- New pricing structures are set in place, leading users to become much more sensitive to the type and quantity of traffic they send and receive.

- The Internet routers switch from their present "first come, first serve" scheduling algorithm for servicing packets to methods that attempt to more equably share resources among different connections (such as Fair Queueing, discussed by Demers, Keshav and Shenker, 1990).

- "Native" multicast becomes widely deployed. Presently, Internet multicast is built on top of unicast "tunnels," so the links traversed by multicast traffic are considerably different from those that would be taken if multicast were directly supported in the heart of the network. And/or: the level of multicast audio and video traffic explodes, as it appeared poised to do a few years ago.

- The Internet deploys mechanisms for supporting different classes and qualities of service (Zhang et al., 1993). These mechanisms would then lead to different connections attaining potentially much different performance than they presently do, with little interaction between traffic from different classes.

- Web-caching becomes ubiquitous. For many purposes, Internet traffic today is dominated by World Wide Web connections. Presently, most Web content is available from only a single place (server) in the network, or at most a few such places, which means that most Internet Web connections are "wide-area," traversing geographically and topologically large paths through the network. There is great interest in reducing this traffic by widespread deployment of mechanisms to support caching copies of Web content at numerous locations throughout the network. As these efforts progress, we could find a large shift in the Internet's dominant traffic patterns towards higher locality and less stress of the wide-area infrastructure.

- A new "killer application" comes along. While Web traffic dominates today, it is vital not to then make the easy assumption that it will continue to do so tomorrow. There are many possible new applications that could take its place (and surely some unforeseen ones, as was the Web only a few years ago), and these could greatly alter how the network tends to be used. One example sometimes overlooked by serious-minded researchers is that of multi-player gaming: applications in which perhaps thousands or millions of people use the network to jointly entertain themselves by entering into intricate (and bandwidth-hungry) "virtual realities."

Obviously, some of these changes will have no effect on some simulation scenarios. But one often does not know *a priori* which can be ignored, so careful researchers must conduct a preliminary analysis of how these and other possible changes might undermine the relevance of their simulation results.

6 Coping Strategies

So far we have focused our attention on the various factors that make Internet simulation a demanding and difficult endeavor. In this section we discuss some strategies for coping with these difficulties.

6.1 The search for invariants

The first observation we make is that, when faced with a world in which seemingly everything changes beneath us, any "invariant" we can discover then becomes a rare piece of bedrock on which we can then attempt to build. By the term invariant we mean some facet of Internet behavior which has been empirically shown to hold in a very wide range of environments.

Thinking about Internet properties in terms of invariants has received considerable informal attention, but to our knowledge has not been addressed systematically. We therefore undertake here to catalog what we believe are promising candidates:

- Longer-term correlations in the packet arrivals seen in aggregated Internet traffic are well described in terms of "self-similar" (fractal) processes. To those versed in traditional network theory, this invariant might appear highly counter-intuitive. The standard modeling framework, often termed Poisson or Markovian modeling, predicts that longer-term correlations should rapidly die out, and consequently that traffic observed on large time scales should appear quite smooth. Nevertheless, a wide body of empirical data argues strongly that these correlations remain non-negligible over a large range of time scales (Leland et al., 1994; Paxson and Floyd, 1995; Crovella and Bestavros, 1996).

  "Longer-term" here means, roughly, time scales from hundreds of milliseconds to tens of minutes. On shorter time scales, effects due to the network transport protocols—which impart a great deal of structure on the timing of consecutive packets—are believed to dominate traffic correlations, although this property has not been definitively established. On longer time scales, non-stationary effects such as diurnal traffic load patterns become significant.

  In principle, self-similar traffic correlations can lead to drastic reductions in the effectiveness of deploying "buffers" in Internet routers in order to absorb transient increases in traffic load (Erramilli, Narayan and Willinger, 1996). However, we must note that the network research community remains divided on the practical impact of self-similarity (Grossglauser and Bolot, 1996). That self-similarity is still finding its final place in network modeling means that a diligent researcher conducting Internet simulations must not *a priori* assume that its effects can be ignored, but must instead consider how to incorporate self-similarity into any traffic models used in a simulation.

  Unfortunately, accurate synthesis of self-similar traffic remains an open problem. A number of algorithms exist for synthesizing exact or approximate sample paths for different forms of self-similar processes. These, however, solve only one part of the problem, namely how to generate a specific instance of a set of longer-term traffic correlations. The next step—how to go from the pure correlational structure, expressed in terms of a time series of packet arrivals per unit time, to the details of exactly *when* within each unit of time each individual packet arrives—has not been solved. Even once addressed, we still face the difficulties of packet-level simulation vs. source-level simulation discussed in §4. In this regard, we note that Willinger et al. (1995) discuss one promising approach for unifying link-level self-similarity with specific source behavior, based on sources that exhibit ON/OFF patterns with durations drawn from distributions with heavy tails (see below).

- Network user "session" arrivals are well-described using Poisson processes. A user session arrival corresponds to the time when a human decides to use the network for a specific task. Examples are remote logins and the initiation of a file transfer (FTP) dialog. Unlike the packet arrivals discussed above, which concern when individual packets appear, session arrivals are at a much higher level; each session will typically result in the exchange of hundreds of packets. Paxson and Floyd (1995) examined different network arrival processes and found solid evidence supporting the use of Poisson processes for user session arrivals, providing that the rate of the Poisson process is allowed to vary on an hourly basis. (The hourly rate adjustment relates to another invariant, namely the ubiquitous presence of daily and weekly patterns in network traffic.) They also found that slightly finer-scale arrivals, namely the individual network "connections" that comprise each session, are *not* well described as Poisson, so for these we still lack a good invariant on which to build.

- A good rule of thumb for a distributional family for describing connection sizes or durations is log-normal. Paxson (1994b) examined random variables associated with measured connection sizes and durations and found that, for a number of different applications, using a log-normal with mean and variance fitted to the measurements generally describes the distribution as well as previously recorded empirical distributions. This finding is beneficial because it means that by using an analytic description, we do not sacrifice significant accuracy over using an empirical description; but, on the other hand, the finding is less than satisfying because Paxson also found that in a number of cases, the fit of neither model (analytic or empirical) was particularly good, due to the large variations in connection characteristics from site-to-site and over time.

- When characterizing distributions associated with network activity, expect to find heavy tails. By a heavy tail, we mean a Pareto distribution with shape parameter $\alpha < 2$. These tails are surprising because the corresponding Pareto distribution has infinite variance. (Some statisticians argue that infinite variance is an inherently slippery property—how can it ever be verified? But then, independence can never be proven in the physical world, either, and few have difficulty accepting its use in modeling.) However, the evidence for heavy tails is very widespread, including CPU time consumed by Unix processes; sizes of Unix files, compressed video frames, and World Wide Web items; and bursts of Ethernet and FTP activity.

- Finally, Danzig and colleagues (1992) found that the pattern of network packets generated by a user typing at a keyboard has an invariant distribution. Subsequently, Paxson and Floyd (1995) confirmed this finding, and identified the distribution as having both a Pareto upper tail and a Pareto body.

6.2 Carefully exploring the parameter space

Another fundamental coping strategy is to judiciously explore the parameter space relevant to the simulation. Because the Internet is such a heterogeneous world, the results of a single simulation based on a single set of parameters are useful for only one thing, namely determining whether the simulated scenario exhibits a show-stopping problem. As one Internet researcher has put it, "If you run a single simulation, and produce a single set of numbers (e.g., throughput, delay, loss), and think that that single set of numbers shows that your algorithm is a good one, then you haven't a clue." Instead, one must analyze the results of simulations for a wide range of parameters.

Selecting the parameters and determining the range of values through which to step them form another challenging problem. The basic approach is to hold all parameters (protocol specifics, how routers manage their queues and schedule packets for forwarding, network topologies and link properties, traffic mixes, congestion levels) fixed except for one element, to gauge the sensitivity of the simulation scenario to the single changed variable. One rule of thumb is to consider orders of magnitude in parameter ranges (since many Internet properties are observed to span several orders of magnitude). In addition, because the Internet includes non-linear feedback mechanisms, and often subtle coupling between its different elements, sometimes even a slight change in a parameter can completely change numerical results (see Floyd and Jacobson, 1992, for example).

In its simplest form, this approach serves only to identify elements to which a simulation scenario is sensitive. Finding that the simulation results do *not* change as the parameter is varied does not provide a definitive result, since it could be that with other values for the fixed parameters, the results *would* indeed change. However, careful examination of *why* we observe the changes we do may in turn lead to insights into fundamental couplings between different parameters and the network's behavior. These insights in turn can give rise to new invariants, or perhaps "simulation scenario invariants," namely properties that, while not invariant over Internet traffic in general, are invariant over an interesting subset of Internet traffic.

7 The VINT Project

The difficulties with Internet simulation discussed above are certainly daunting. In this section we discuss a collaborative effort now underway in which a number of Internet researchers are attempting to significantly elevate the state-of-the-art in Internet simulation.

The VINT project (http://netweb.usc.edu/vint/) is a joint project between USC/ISI, Xerox PARC, LBNL, and UCB to build a multi-protocol simulator, using the UCB/LBNL network simulator "ns" (http://www-mash.cs.berkeley.edu/ns/). The key goal is to facilitate studies of scale and protocol interaction. The project will incorporate libraries of network topology generators and traffic generators, as well as visualization tools such as the network animator "nam" (http://www-mash.cs.berkeley.edu/ns/nam.html).

Most network researchers use simulations to explore a single protocol. However, protocols at different layers of the protocol stack can have unanticipated interactions that are important to discover before deployment in the Internet itself. One of the goals of the VINT project is building a multi-protocol simulator that implements unicast and multicast routing algorithms, transport and session protocols, reservations and integrated services, and application-level protocols such as HTTP. In addition, the simulator already incorporates a range of link-layer topologies and scheduling and queue management algorithms. Taken together, these enable research on the interactions between the protocols and the mechanisms at the various network layers.

A central goal of the project is to study protocol interaction and behavior at significantly larger scales that those common in network research simulations today. The emphasis on how to deal with scaling is not on parallel simulation techniques to speed up the simulations (though these would be helpful, too), but to allow the researcher to use *different levels of abstraction* for different elements in the simulation.

The hope is that VINT can be used by a wide range of researchers to share research and simulation results and to build on each other's work. This has already begun for some areas of Internet research, with researchers posting ns simulation scripts to mailing lists to illustrate their points during discussions. If this *lingua franca* extends to other areas, then VINT might play an important role in abetting and systematizing Internet research.

8 The Role Of Simulation

We finish with some brief comments on the role of simulation as a way to explore how the Internet behaves, versus experimentation, measurement, and analysis. While in some fields the interplay between these may be obvious, Internet research introduces some unusual additions to these roles.

Clearly, all four of these approaches are needed, each playing a key role. All four also have weaknesses.

Measurement is needed for a crucial "reality check." It often serves to challenge our implicit assumptions. Indeed, while we personally have conducted a number of measurement studies, we have never conducted one that failed to surprise us in some fundamental fashion.

Experiments are crucial for dealing with implementation issues, which can at first sound almost trivial, but often wind up introducing unforeseen complexities. Experimentation also plays a key role in exploring new environments before finalizing how the Internet protocols should operate in those environments.

One problem Internet research suffers from that most other fields do not is the possibility of a "success disaster"—designing some new Internet functionality that, before the design is fully developed and debugged, escapes into the real world and multiplies there due to the basic utility of the new functionality. Because of the extreme speed with which software can propagate itself to endpoints over the network, it is not at all implausible that the new functionality might spread to a million computers within a few months. Indeed, the HTTP protocol used by the World Wide Web is a perfect example of a success disaster. Had its designers envisioned it in use by virtually the entire Internet—and had they explored the corresponding consequences with experiments, analysis or simulation—they would have significantly altered its design, which in turn would have led to a more smoothly operating Internet today.

Analysis provides the possibility of exploring a model of the Internet over which one has complete control. Analysis plays a fundamental role, because it brings with it greater understanding of the forces at play. It carries with it, however, the serious risk of using a model simplified to the point where key facets of Internet traffic have been lost, in which case any ensuing results are useless (though they may not appear to be so!).

Simulations are complementary to analysis, not only by providing a check on the assumptions of the model and on the correctness of the analysis, but by allowing exploration of complicated scenarios that would be either difficult or impossible to analyze. (Simulations can also play a vital role in helping researchers to develop intuition.) The complexities of Internet topologies and traffic, and the central role of adaptive congestion control, make simulation the most promising tool for addressing many of the questions of Internet traffic dynamics. As we have illustrated, there does not exist a single suite of simulations sufficient to prove that a proposed protocol performs well in the Internet environment. Instead, simulations play the role of examining particular aspects of proposed protocols, and, more generally, particular facets of Internet behavior.

For some topics, the simplest scenario that illustrates the underlying principles is often the best. In this case the researcher can make a conscious decision to abstract away all but what are judged to be the essential components of the scenario under study. As the research community begins to address questions of scale, however, the utility of small, simple simulation scenarios is reduced, and it becomes more critical for researchers to address questions of topology, traffic generation, and multiple layers of protocols.

Finally, we hope with this discussion to spur, rather than discourage, further work on Internet simulation. The challenge, as always, is to keep the insight and the understanding, but also the realism.

Acknowledgments

Our thanks to Kinh Tieu for his careful reading of an earlier draft.

References

Calvert, K., M. Doar, and E. W. Zegura. 1997. Modeling Internet topology. *IEEE Communications Magazine* 35:160–163.

Crovella, M. and A. Bestavros. 1996. Self-similarity in World Wide Web traffic: evidence and possible causes. *Proceedings of SIGMETRICS '96*.

Cunha, C., A. Bestavros, and M. Crovella. 1995. Characteristics of WWW client-based traces. Technical Report TR-95-010, Boston University Computer Science Department.

Danzig, P., S. Jamin, R. Cáceres, D. Mitzel, and D. Estrin. 1992. An empirical workload model for driving wide-area TCP/IP network simulations. *Internetworking: Research and Experience* 3:1–26.

Demers, A., S. Keshav, and S. Shenker. 1990. Analysis and simulation of a fair queueing algorithm. *Internetworking: Research and Experience* 1:3–26.

Erramilli, A., O. Narayan, and W. Willinger. 1996. Experimental queueing analysis with long-range dependent packet traffic. *IEEE/ACM Transactions on Networking* 4:209–223.

Floyd, S., and V. Jacobson. 1992. On traffic phase effects in packet-switched gateways. *Internetworking: Research and Experience* 3:115–156.

Grossglauser, M., and J-C. Bolot. 1996. On the relevance of long-range dependence in network traffic. *Proceedings of SIGCOMM '96*, pp. 15–24.

Jacobson, V. 1988. Congestion avoidance and control. *Proceedings of SIGCOMM '88*, pp. 314–329.

Leland, W., M. Taqqu, W. Willinger, and D. Wilson. 1994. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking* 2:1–15.

Lottor, M. 1997. *http://www.nw.com/zone/WWW/top.html*.

Paxson, V. 1994a. Growth trends in wide-area TCP connections. *IEEE Network* 8:8–17.

Paxson, V. 1994b. Empirically-derived analytic models of wide-area TCP connections. *IEEE/ACM Transactions on Networking* 2:316–336.

Paxson, V., and S. Floyd. 1995. Wide-area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking* 3:226–244.

Paxson, V. 1996. End-to-end routing behavior in the Internet. *Proceedings of SIGCOMM '96*, pp. 25–38.

Paxson, V. 1997. Automated packet trace analysis of TCP implementations. *Proceedings of SIGCOMM '97*. To appear.

Willinger, W., M. Taqqu, R. Sherman, and D. Wilson. 1995. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *Proceedings of SIGCOMM '95*, pp. 100–113.

Zhang, L., S. Deering, D. Estrin, S. Shenker, and D. Zappala. 1993. RSVP: a new resource reservation protocol. *IEEE Network*. 7:8-18.