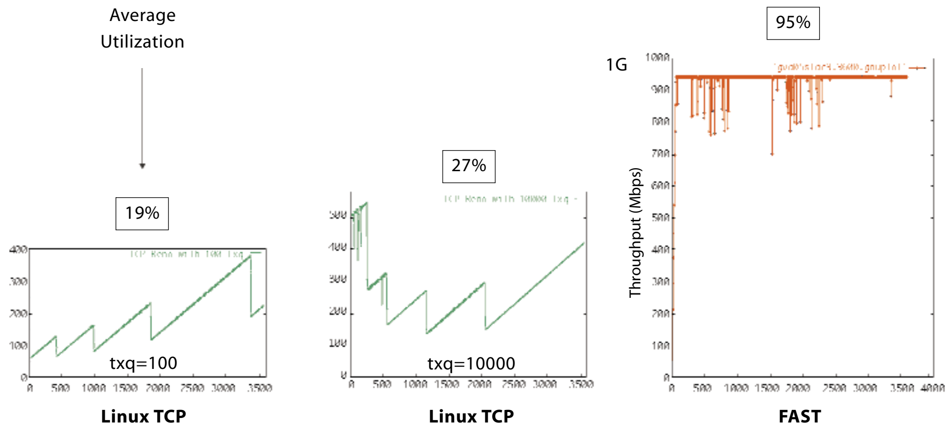
Today's Internet is Just a Parking Lot: New Algorithms Pave the Way for the Real Superhighway—Steven Low and the FAST Team

Imagine driving in the following manner. You fix your eyes twenty feet in front of your car. When you see no obstacle in sight, you slowly increase your speed. As soon as anything comes into sight, you slam on the brakes. Repeat as many times as necessary to reach your destination.

This is not a bad strategy in a parking lot, but it is not how you would want to drive on the autobahn, or on any high-speed roadway. When we enter a highway, we look near and far, front and rear, to sense the traffic flow around us and then converge quickly to the right speed. When sending and receiving information over the Internet, the current strategies used to control transfer speed are still designed for parking lot driving, while the infrastructure is being upgraded to superhighway mode. The FAST project at Caltech, led by Professor Steven Low, is changing that.

Here is the current state of affairs. The algorithm that controls the sending speed is called the congestion control algorithm. It is a distributed algorithm designed to share the Internet among hundreds of millions of users. It consists of two sub-algorithms. One is a host algorithm, implemented by the Transmission Control Protocol (TCP) that modifies the sending rate in response to the amount of congestion in the path from source to destination. The second is a queue-management (QM) algorithm that implicitly or explicitly feeds back congestion information to the hosts. All currently deployed TCP algorithms are based on a scheme developed at Berkeley 15 years ago when most parts of the Internet could barely carry the traffic of a single voice call—and when the general public had no knowledge of the Internet's existence, let alone surfed the web. Since the mid 1990s, researchers have realized that this algorithm cannot scale to any future network that must be able, for instance, to carry 1.5 million concurrent voice calls. The lack of a theoretical framework to understand the *underlying structure of the general problem* led to a tremendous variety of *ad hoc* tinkering with the TCP and QM algorithms, with limited success.

During the past five years, a rigorous theory of congestion control has started to emerge, based on work done at Cambridge University, the University of Melbourne, Caltech, UCLA, UIUC, and the University of Massachusetts. The theory covers the equilibrium and dynamic structure of large-scale networks under end-to-end control. One of the key components in the theory is Low's idea of interpreting the TCP/QM pair as a distributed primal-dual algorithm carried out over the Internet by hosts and routers in the form of congestion control, in order



to solve a global mathematical optimization problem. The iteration on the primal variable is carried out by TCP while the iteration on the dual variable is carried out by QM. The underlying optimization problem determines the equilibrium properties of the network such as performance, throughput, delay, packet loss, and fairness in resource allocation. It explains some intriguing phenomena observed empirically and provides practical guidelines for sizing network buffers.

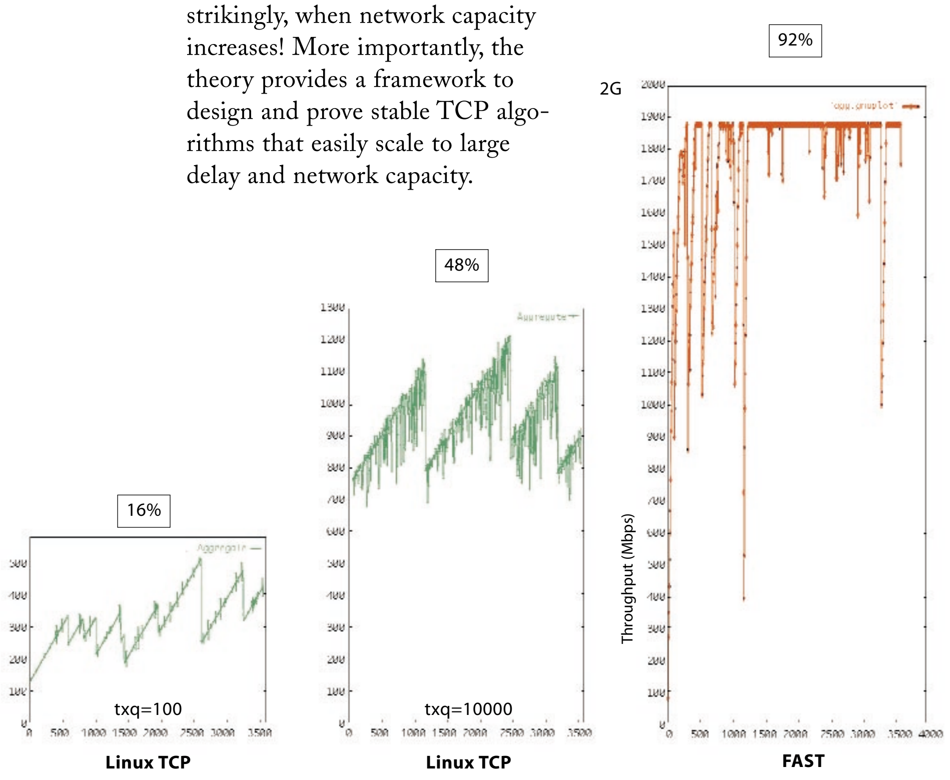
With Caltech Professor John Doyle and Caltech alumnus and UCLA Professor Fernando Paganini (MS '92, PhD '96), Low has shown that the *current* TCP algorithm becomes unstable when feedback delay increases, and more strikingly, when network capacity increases! More importantly, the theory provides a framework to design and prove stable TCP algorithms that easily scale to large delay and network capacity.

With these new theoretical insights as a guiding light, Low's Networking Lab designed a TCP congestion control algorithm, called FAST TCP, that maintains high performance and is fair and stable. It was implemented in the Linux operating system. With Caltech Professor Harvey Newman, they demonstrated FAST TCP in November 2002 at the Supercomputing Conference in Baltimore, Maryland. The partners involved for this demonstration included Caltech's Center for Advanced Computing Research (CACR), Stanford Linear Accelerator Center (SLAC), European Organization for Nuclear Research (CERN), with support from DataTAG, StarLight, TeraGrid, Cisco, and Level(3).

The current TCP typically achieves an average throughput of 266 Mbps, averaged over an hour (a single TCP/IP flow between Sunnyvale, California and CERN in Geneva, Switzerland over a distance of 10,037 kilometers). This represents an efficiency of just 27 percent. Using the standard packet size that is supported throughout today's networks, the FAST TCP sustained an average throughput of 925 Mbps and an efficiency of 95 percent—a 350 percent improvement—under the same experimental conditions. With 10 concurrent TCP flows, FAST achieved an unprecedented speed of 8,609 Mbps, at 88 percent efficiency. More importantly, the FAST protocol sustained these speeds using standard packet size stably over an extended period on shared networks in the presence of background traffic, making it "backwards compatible" and adaptable for deployment on the world's high-speed production networks.

Low and his team are working with academic and industry partners to further test the FAST TCP in real applications. The first users will probably be those in academia requiring terabyte data transfers on demand, high-energy physics (HEP) research groups, for instance. But stay tuned! This new superhighway may be making its way to your neighborhood next. ENG

Professor Steven Low is Associate Professor of Computer Science and Electrical Engineering at Caltech.



The figure above shows throughput traces for FAST and Linux TCP with percentage utilization. The x-axis is time (sec), the y-axis is aggregate throughput (Mbps). The FAST traces are in red. The txq number refers to the size of buffer memory in the Linux kernel used to mitigate the mismatch in the network card's transmission rate and the CPU processing of packets.

For further information visit:
netlab.caltech.edu/FAST