Ultrascale Network Protocols for Computing and Science in the 21st Century *

Julian J. Bunn
Center for Advanced Computing Research (CACR), Caltech
julian@cacr.caltech.edu

John C. Doyle
CDS, EE and BE, Caltech
doyle@cds.caltech.edu

Steven H. Low
CS and EE, Caltech
slow@caltech.edu

Harvey B. Newman
Physics, Caltech
newman@hep.caltech.edu

Steven M. Yip
Cisco Systems
syip@cisco.com

http://netlab.caltech.edu/FAST/

September 12, 2002

Abstract

There is a clear and urgent need for ultrascale networking, that provides more than 100 Gbps of throughput end-to-end to transfer Petabyte files, to support the next generation of scientific computing and discoveries. Rapid advances in computing, communication and storage technologies will provide the required raw capacities. The application of these technologies to effective working environments is underway through the revolutionary development of data-intensive Grid systems on a global scale. We must, however, reexamine and redesign the network control protocols that are at the foundation of these transformative trends in science and computer science, to enable us to scale up the networks, and the Grid systems built upon them, by orders of magnitude in the decade ahead.

*White paper to DoE's USS (Ultrascale Simulation for Science) initiative.

1 Motivation

There is a clear and urgent need for multi-Gigabit networks in the scientific community, today, and a need for ultrascale networks in the near future that provide more than 100 Gbps of sustainable throughput end-to-end to transfer Petabyte files. Moreover, continued advances in computing, communication, and storage technologies, combined with the development of national and global Grid systems, hold the promise of providing the required capacities, and an effective environment for the next generation of scientific discoveries. In the next section, we will explain why the current control protocol to share network resources cannot scale to this regime.

1.1 Demand for ultrascale networking

The HENP (High Energy and Nuclear Physics) community has a long tradition of pushing computing and networking technologies to their limits, in production environments. This trend has accelerated in the last few years both due to the Petabytes (10^{15} bytes) of data acquired, stored, distributed and processed by the worldwide HENP collaborations, and due to the development of Data Grids, which aim to make the data available rapidly and transparently to scientists around the globe. Experiments now underway at SLAC, Fermilab and Brookhaven are already accumulating Petabyte datasets. The next generation of particle physics experiments now under development, due to begin operation in 2007 at CERN in Geneva, will deal with data volumes of tens of Petabytes (in 2007–2008) to Exabytes (10^{18} bytes) in the decade following. This will impose tremendous new demands on computing, communication and storage technologies.

A current example illustrating the data and computationally intensive character of HENP problems encountered by research teams is the search for Higgs particles at the LHC (the Large Hadron Collider at CERN). A full optimization of the separation of the Higgs discovery signal from potentially overwhelming backgrounds is estimated to require 10^8 fully simulated and reconstructed background events, drawn from 10^{11} generated events (sets of simulated four vectors) using loose pre-selection criteria. The processing requirement is approximately 10^6 CPU-days, or 10,000 of today's fastest processors used round the clock for three to four months. The data resulting from this study will be on the order of 200–400 Terabytes. This implies a need to transfer 2–4 Terabytes per day produced in bursts, which will take 0.5–1 hour of transfer time per day at a throughput of 1 Gbyte/sec end-to-end over the wide area.

Projecting a few years ahead, systems with 10-100 TFlops will be available (the Earth Simulator that was unveiled earlier this year is 36 TFlops) and transfers of 1 Petabyte files will not be uncommon. This requires a global ultrascale network that provides more than 100 Gbps throughput end-to-end. As the decade wears on, this requirement is expected to grow to the Terabit/sec (Tbps) range as the short data "bursts" used by scientists and engineers progress from 1 to 10, and to the 100 Terabyte scale in some cases.

1.2 Enabling technologies

The ability to scale silicon technology improves the performance of the devices and decreases their cost, both at an exponential rate. The number of transistors in an integrated circuit has increased by eight orders of magnitude, from two transistors in the 1960s to more than 100 million today. Over the same period, gates have become 1,000 times faster and consume 10,000 times less power.

This drastic increase in computing power is more than matched by the advances in communication capacity. The capacity of telephone and data backbone networks has increased by more than three orders of magnitude over the last decade to 10Gbps (the limit of current production-line electronics) on fiber links. The deployment of wave-division multiplexers (WDM) has overcome this electronic speed limit by transporting tens of 10Gbps channels on the same fiber, further increasing the link capacity to several hundred Gbps in the last few years. In the future, it is anticipated that optical cross-connects

and other transmission technologies such as soliton sources will create pure optical networks where Tbps bandwidth will be common.

Storage technology has also been keeping up, growing from about 2 kbits per square inch in the mid-1950s when disk drives were first invented to 1 Terabit per square inch demonstrated by IBM in mid-2002: an increase by nine orders of magnitude.

In parallel to, and driven by, this dramatic increase in performance and decrease in cost of computing, communication, and storage is the spectacular growth of the Internet, from connecting four hosts in 1969 to hundreds of million hosts today, an increase by eight orders of magnitude.

Modeling studies and extrapolations of the rapid advances in computing, communication, and storage technologies show that sufficient capacity will be available for the new generation of scientific computing. The key challenge we face, and intend to overcome, is that our current network control and resource sharing algorithms cannot scale to this regime. Without profound developments in scalable protocols, we cannot build the networks we require to fulfill this vision.

2 Protocol scalability

We now explain why the paradigm for sharing network resources in the current Internet – statistical multiplexing with *proper* end-to-end flow control – should be preserved in the future networks, the difficulties in scaling the current TCP, and how to address them.

2.1 End-to-end flow control

The most important difference between a packet-switched (or packet) network and a circuit-switched (or circuit) network is that, in the latter, when a connection is established between a source and a destination, network resources (e.g., time slots in time-division multiplexed systems, frequency slots in frequency-division multiplexed systems) are reserved along the path for its exclusive use for the duration of the connection. The traditional telephone network is an example of circuit network. The fixed rate allocation simplifies the control of the system and the provisioning of quality of service (QoS). Circuit networks are suitable for applications that generate traffic at a fixed rate, such as uncoded voice. Traffic is regulated at the connection level through connection admission control, which decides whether or not a new connection request is granted, depending on, e.g., the availability of resources at the time of the request. End-to-end flow control, that adapts transmission rates of admitted sources to changes in the availability of network resources along their paths, is unnecessary.

In a packet network, in contrast, a path may be established between a source and its destination during the connection setup phase, but no bandwidth or buffer resources are reserved. Rather, these resources are shared by all connections on demand. This is ideal for applications that generate bursty traffic, or that have an elastic bandwidth requirement and hence can adapt their rates to network or receiver congestion. Statistical multiplexing, however, makes the control and provisioning of QoS harder. It is difficult to characterize the resource requirements of such applications, and hence connection admission control is rarely implemented in packet networks. Since the number of connections in the network is not controlled, their source rates must be dynamically regulated to avoid overwhelming the network or the receivers. This is the purpose of end-to-end flow control.

It is possible to stream a Terabyte file at a large fixed rate over a reserved "circuit", without the need for end-to-end flow control. This circuit-switching approach, however, is inefficient because, unlike uncoded voice, there is not a natural rate to reserve for bulk transfers. Indeed, bulk transfers are inherently elastic, in that they can take full advantage of increased bandwidth or reduce rate to accommodate other traffic. It is extremely likely that the available bandwidth will fluctuate during the (long) lifetime of a bulk transfer, due to arrivals and departures of other transfers. It is therefore much more efficient to allow the transfers to share bandwidth dynamically, in future networks as they do in

current ones. Statistical multiplexing, together with end-to-end flow control, is a key factor that has enabled an explosive set of applications to share the Internet efficiently.

Indeed, traffic generated by scientific computing applications is ideal for end-to-end flow control because of its extreme heavy-tailed nature. Extensive research has shown that Internet traffic is heavy-tailed, and this can be traced to the heavy-tailed distribution of file sizes. Heavy-tailed file sizes in turn can be shown to be an inevitable outcome of engineering design. An important implication of such traffic is that, even though most files are small ("mice"), most packets belong to huge files ("elephants"). End-to-end flow control aims to control the elephants to maximally utilize network bandwidth, in a way that leaves the network queues mostly empty so that mice can fly through the network without much delay or loss.

HENP applications, for example, produce gigantic elephants, making the tail of the file size distribution heavier. The heavier the tail, the better end-to-end flow control works, because the duration of typical connections will be large compared with the convergence time of the control mechanism.

2.2 Scalability problems of TCP

A breakthrough that has allowed the Internet to expand by four orders of magnitude in size and five orders of magnitude in backbone speed in the last 15 years was the invention in 1988 by Van Jacobson of the TCP (Transmission Control Protocol) end-to-end flow control algorithm. TCP is a distributed and asynchronous algorithm to share network resources among competing users. It has been carrying more than 90% of the Internet traffic and is instrumental in preventing the Internet from congestion collapse while the Web exploded in the 1990s.

The current TCP however cannot operate efficiently in the bandwidth regime of the future, due to serious equilibrium and stability problems that lead to wildly oscillating transmission rates and even erratic network behavior at high speed. For instance, the current protocol uses packet loss as a congestion measure, which has three difficulties at high speed. First, losses must be extremely rare to support the window size of ultrascale networking. For example, to achieve a throughput of 50 Gbps over a distance with 200 ms round-trip delay using a packet size of 100 kbits will require a loss probability on the order of 10^{-10}. If the current packet size of 12 kbits is used, then the loss probability needs to be on the order of 10^{-12}. This can be difficult to achieve. Second, even if this loss probability is achieved, it is an extremely noisy feedback signal for the sources to reliably use for control. Finally, since TCP must induce loss in order to estimate the available bandwidth, however rare losses are, when they *inevitably* occur, they occur in bursts, increasing the likelihood of timeout and underutilization of the network. Besides the problem with using loss probability for control, the way TCP adapts its rate induces instability at high speed, making wild oscillations unavoidable. These problems must be overcome in order to scale TCP to the multi-Gbps long-distance regime.

TCP's basic flaws, combined with other factors such as limited network access speeds and ill-suited default parameter settings in PCs, have been the culprit behind the observed under-utilization of network resources on major backbones throughout the world. In many countries including the US, these limitations have had a profound economic impact through reduced productivity and lower efficiency of operation of networked systems. As 1-10 Gbps Ethernet, 10 Gbps and faster network backbones, and improved operating systems and default settings are becoming the norm, the main impediment to progress will soon be TCP's lack of scalability.

2.3 Scalable protocols

Exciting advances have been made in the last couple years on understanding the equilibrium and dynamic behavior of large networks, such as the Internet. There is now a preliminary theory both to analyze the scalability problems of existing protocols, and to guide the design of new protocols that can in principle scale to arbitrary capacity and delay. We are currently implementing these advances and aim

to demonstrate by the end of this year the feasibility of end-to-end flow control in a *dynamic* environment in the Gbps range over links with 100-200 ms round-trip delay, corresponding to transcontinental and intercontinental distances. Next year we plan to extend these developments to the 10 Gbps range.

The preliminary theory applies to any flow control scheme that works within the decentralization constraints inherent in a large network. It therefore applies not only to TCP, but also to RTP/RTCP or any end-to-end flow control algorithm implemented on top of UDP.

Despite these advances, much work remains, both on the theoretical and practical fronts. For instance, the current theory is only local and describes the network behavior around equilibrium. A global theory for a distributed network with delay is important. The current implementation effort is restricted to modification of TCP. With changes in active queue management algorithms in routers as well, we will be able to achieve both high utilization and low loss and queueing delay at arbitrary network capacity and size. Other unexplored issues of both theoretical and practical importance include the interaction of control algorithms across protocol layers and across generations in the evolution path.

3 Conclusion

The development of robust and stable ultrascale networking, at 100 Gbps and higher speeds in the wide area, is critical to support the new generation of ultrascale computing and Petabyte to Exabyte datasets that promise to drive discoveries in fundamental and applied sciences of the next decade. In order to achieve these goals we must, however, reexamine, and, when necessary, redesign, the control protocols that manage these resources, based on a sound and rigorous theoretical foundation, and develop the practical means to scale their capabilities up by orders of magnitude, to meet the demands of future networks and applications.