Fluid Multipath Transport for Big Data Transfers

Armando Caro
Raytheon BBN Technologies
acaro@bbn.com

***Abstract*—Data management for storage, collaboration and computation remains a critical issue on the ESnet. We present several research challenges for the network and transport layers that aim to improve efficiency of Big Data transfers. Our focus is on enabling technologies that would provide the building blocks for a fluid multipath transport system that fully utilizes network path resources.**

I. The Problem

ESnet is a high-speed network that connects experimental and computational facilities to support thousands of DOE scientists worldwide. Experiments produce vast amounts of data, some of which currently becomes "dark data" due to antiquated approaches for storing and transporting data. Making data readily available to all researchers remains a challenge.

Big Data cloud solutions are applicable and can help address storage and computational needs, but they generally fail to address networking issues with Big Data transfers. In the commercial world, Big Data systems usually accumulate large amounts of data over time from multiple sources (e.g., social media posts) and then attempt to minimize how much data is transferred over the network during computation. But in the scientific world, the need to transfer a single voluminous blob or a high speed stream from a single source is not uncommon. Data generated in an experimental facility may be transferred in real-time to one or more computational facilities, or existing data may be replicated to other locations for accessibility and archival reasons. In any case, current communications protocols are inefficient at transferring Big Data over a network.

The goal of a transport protocol is to fully utilize network path resources from sender to receiver fairly among all pairs of end-hosts sharing part of the same path. Traditionally, the approach used is a closed-loop, end-to-end system that relies on AIMD probing algorithms that estimate available path resources and respond to network congestion in a reactive way. AIMD has been shown to achieve fairness and stability, but suffers from reactive response to congestion and packet loss, burstiness, RTT-unfairness, slow flow completion time, and the ability to cause congestion/loss. Decades of research have made incremental improvements by optimizing AIMD for improved feedback response and instrumenting techniques that provide explicit and/or faster feedback to the sender to overcome the slow, end-to-end feedback loop. However, the endpoint-based closed-loop congestion control approach of TCP and the like is bound to be inefficient due to decisions only made at the endpoints.

II. Research Challenges

The state-of-the-art is far from maximizing available bandwidth fairly for sustained periods of time. The ultimate goal of AIMD (and TCP) is to estimate the capacity of the lowest-throughput link along the delivery path and transmit traffic according to that. But this does not fully utilize other links in the path. Ideally, we want to maximize utilization of the highest-throughput link. Other lower-throughput links along the path would be supplemented by multipath routing across multiple links concurrently to sustain the same aggregate throughput from source to destination.

Multipath TCP (MPTCP) uses multipath congestion control algorithms to exploit multiple end-to-end paths for sending traffic and potentially achieving higher throughput, but it's restriction to end-to-end paths and utilization of AIMD cause it to exhibit the same inefficiencies as normal TCP. Multipath routing has been used for traffic engineering and load-balancing, but congestion control and sending rates are still regulated to the endpoints. Little work has been done to combine the benefits of multipath routing and multipath congestion control to improve overall resource utilization.

Psaras et al. propose in *HotNets 2014* to use backpressure routing to engage additional paths when a bottleneck link is encountered along a path. Their approach uses in-network caching to avoid losses during transitions when a bottleneck link cannot be alleviated with additional paths and the sender is being notified that the sending rate must be reduced. This approach shows promise, but the in-network caches introduce complexity and cost that should not be necessary. A transport protocol that is tolerant of packet loss and requires no retransmissions would enable sending rate transitions without requiring in-network caches to avoid data loss.

QUIC is an experimental transport protocol that Google is designing to reduce application experienced latency. One technique is to use packet-level forward error correction to reduce the number of retransmissions, thereby lowering latencies. Although this a step in the right direction, QUIC uses a fixed code rate which limits the error recovery flexibility. Fountain codes (or rateless erasure codes) may provide the flexibility needed to design a retransmission-free reliable transport protocol that focuses on transmission rate and global fairness based on network and/or peer-endpoint feedback.

To summarize, we argue that a fluid multipath transport system can potentially maximize throughput and minimize latency of Big Data transfers. Such a system likely involves significant changes to the network and transport layers, and a tighter coupling between them. The research challenges at the transport layer are two-fold. Eliminate (or at least minimize) retransmissions using fountain codes for packet-level erasure correction. Dynamically adjust the sending rate to maximize end-to-end throughput using network layer indicators and end-to-end feedback. At the network layer, the major challenge is computing congestion-free multipath routes that maximize the throughput of transport layer flows.

References

References omitted due to space limitations.