Improving throughput by dynamically adapting concurrency of data transfer

Prasanna Balaprakash1,2, Vitali Morozov3, Rajkumar Kettimuthu4
1 Mathematics and Computer Science Division & 2 Argonne Leadership Computing Facility
Argonne National Laboratory, 9700 South Cass Avenue, Lemont, IL 60439

Abstract
Improving the throughput of data transfer over high-speed long-distance networks has become increasingly difficult and complex. Numerous factors such as varying congestion scenarios, external factors which are hard to characterize analytically, dynamics of underlying transfer protocol contribute to this difficulty. In this study, we consider optimizing memory-to-memory transfer via TCP, where the data is transferred from a source memory to a destination memory using TCP. Inspired by the simplicity and the effectiveness of additive increase multiplicative decrease schemes of TCP variants, we propose a tuning algorithm that can dynamically adapt the number of parallel TCP streams to improve the aggregate throughput of data transfers.

Factors affecting throughput
TCP dynamics
Host cannot keep up with the network bandwidth delay product
Congestion in the bottleneck link due to external traffic

Throughput optimization
Given a source src, destination dst, and size $x$ of the data that need to be transferred, the problem of optimizing the performance of the file transfer can be formulated as follows:

$$\arg \max_{x \in D} \int_{T_{start}}^{T_{end}} f_1(x, t, \delta, \theta_{src}^t, \theta_{dst}^t) dt,$$

where $x$ is a vector of $m$ controllable tuning parameters.

The objective
• $x$ is the size of the data transferred from $t'' - t' = dt$, $\delta$ is the hyperparameter capturing net work condition at time $t$.
• $\theta_{src}^t$ and $\theta_{dst}^t$ are the parameters describing the source and the destination loads at time $t$.
• $T_{start}$ and $T_{end}$ denote the file transfer starting and ending time, respectively.

• The non-negative function $f_1$ is the transfer performance metric, typically throughput, at time $t$, for the parameter configuration $x$.

The network congestion and source-destination bottlenecks can vary with time because of other transfers/workloads can share the same resources.

• Maximum achievable throughput will change over time.
• To take into account this dynamism, solution to the problem consists in maximizing throughput for every $x$ between $T_{start}$ and $T_{end}$.

Congestion link model
• Wide area network consists of sources and destinations connected together via links and routers.
• All of the sources operate TCP-like congestion control algorithms.
• The links and queues along a network path form a 'pipe' that contain packets in flight.
• TCP congestion control is achieved by dynamically adapting the window size according to an additive-increase multiplicative-decrease scheme.
• Source probes the network for spare capacity for additional bandwidth and back-off the number of packets transmitted when congestion is detected.

Goal
Develop an online tuning framework for optimizing file transfers throughputs.

Tuning algorithm
At each control epoch $c$, the following control logic determines $n_{src}$.

$$n_{src} = \begin{cases} n_{src} - 1, & \text{if } n_{src} \leq n_{src} - 1 \text{ and } |\Delta_t| > \epsilon \\ n_{src} + 1, & \text{if } n_{src} > n_{src} - 1 \text{ and } n_{src} - 1 \neq n_{src} - 2 \text{ and } \delta_t \geq \epsilon \\ n_{src} - 1, & \text{if } n_{src} > n_{src} - 1 \text{ and } n_{src} - 1 \neq n_{src} - 2 \text{ and } \delta_t < -\epsilon \\ n_{src} - 1, & \text{otherwise} \end{cases}$$

where,

$$\Delta_t = c_t - \delta_t = \Delta_t + \delta_t \left( n_{src}^{t-1}, \ldots \right)$$

and

$$\delta_t = \frac{\Delta_t}{n_{src} - 1 - n_{src} - 2}$$

• When link or TCP becomes bottleneck, increase the number of streams.
• When host becomes bottleneck, decrease the number of streams.
• No significant change in throughput, do not change the number of streams.

Experimental results
• Source: Argonne, Destination: Univ. of Chicago.
• The peak throughput is 1 Gbps. 125 Mb/s).
• Globus-url-copy from Globus toolkit for memory to memory transfer.
• Artificially introduced congestion in the 1 Gbps link by starting a transfer in the background from source.

Our studies show that results show significant throughput improvement potential under various congestion conditions.

Future investigations
• Algorithmic improvements
• Wide area transfer studies
• Including other tuning parameters
• Disk to disk transfer and other potential parameters
• Implementation in production tools - Congestion and bottleneck analysis

Support for this work was provided through RAMSES project funded by the US Department of Energy’s Office of Advanced Scientific Computing Research (ASCR), under the leadership of Rich Carlson.