

EECS 428 Project

Ian Dimayuga

April 29, 2012

1 Correctness of Simulation

1.1 Exponential Flows

The simulation code specifies an average on time of $\mu = 60$ seconds, and an average off time of $w = 180$ seconds. The total duration of the simulation $S = 2000$ seconds, and the number of unidirectional browser flows was $n = 30$. The total number of bytes transmitted by the exponential flows in one direction was measured to be $T = 111141750$ bytes, or approximately 106 MB.

Theoretically, the total number of bytes transmitted is given by the equation $T = \frac{Sn\mu r}{\mu + w}$, where r is the average on-time rate of 56 kbps.

If we evaluate the equation for T , we get $T = 107520000$ bytes, or 103 MB. This matches the measured value of T . Dividing this by S gives us an average rate of 53760 B/s, or about 0.41 Mbps.

1.2 Pareto Flows

For the Pareto flows, we specify an average on time $\mu = 500$ milliseconds, and an average off time of $w = 60$ seconds. The Pareto on-time rate r was 128 kbps. The total number of pareto bytes measured was $T = 8406450$ bytes, or approximately 8 MB.

If we evaluate the same equation for T , we get $T = 8124298$ bytes, or 8 MB. This matches the measured value of T . Dividing this by S gives us an average rate of 4062 B/s, or about 0.03 Mbps.

1.3 Delayed Acknowledgements

All DelACK timeouts were measured to be 50ms as specified. There are 10905 DelACK timeouts, and 432445 immediate ACKs triggered by data packets. The TCP packet to ACK ratio is still 2.

1.4 Elephants

Given the bandwidth taken up by the Exponential (UDP) flows, TCP packets are left with $10 - r_{exp} = 9.5$ Mbps of bandwidth through the central link. This can be corroborated by measuring the amount of data being ACKed over time. As shown in Figure 1, the data ACKed over time does not change as the first Elephant finishes at 704s. The initial slope of this graph (9.5 Mbps) shows that the link is at full utilization all the way until the second Elephant finishes. There are distinct slope changes when the second and third Elephants finish. Furthermore, the total number of bytes ACKed at the conclusion of the trace is 1.21 GB, which is equal to the sum of the three 400 MB files and the 16 MB of total data in the pareto flows.

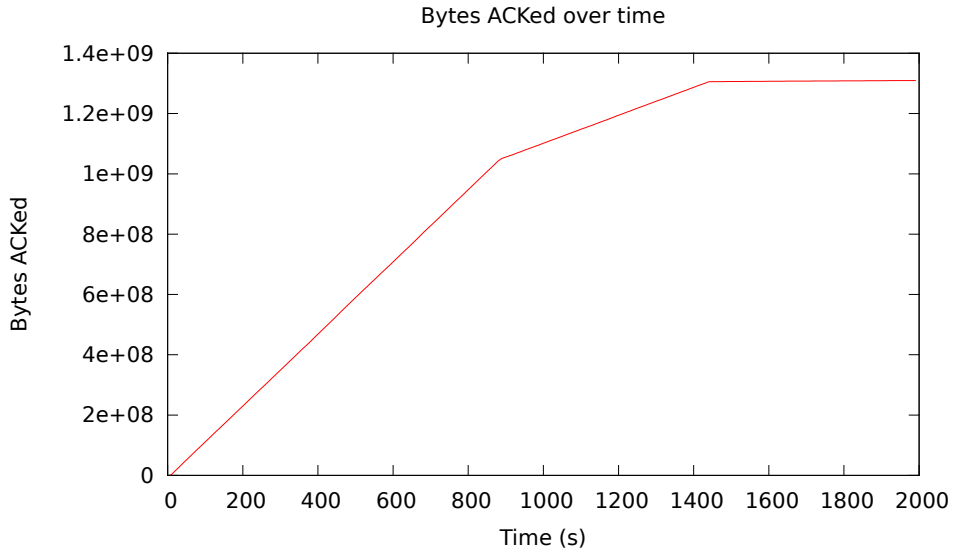


Figure 1: The cumulative number of bytes ACKed at the source, taken every 10 seconds.

2 Elephant Utilization

As shown in figure 2, the total Elephant flow bandwidth usage stays at around 9 Mbps until the second elephant finishes at 885s. However, the third elephant does not take full utilization after this time. This is also indicated by the change in slope in 1.

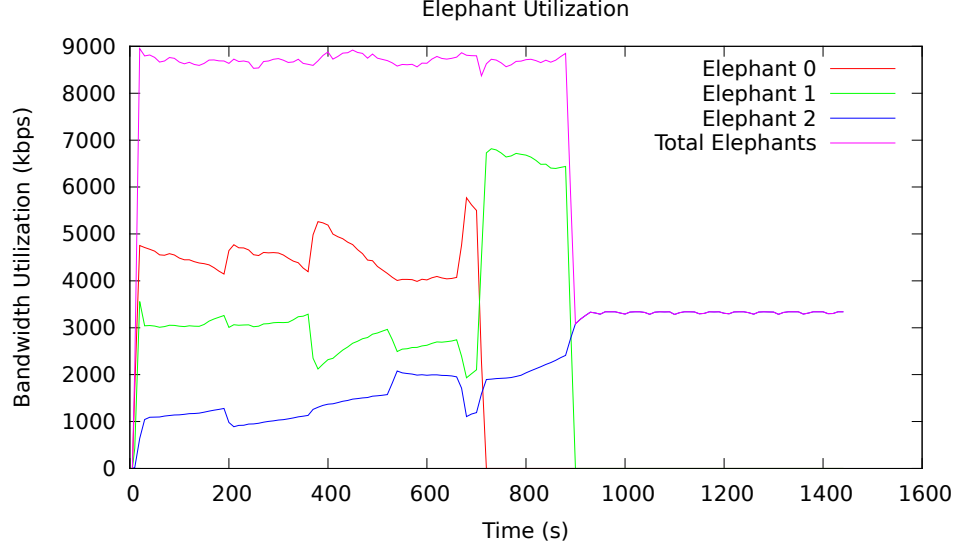


Figure 2: The bandwidth utilization of each elephant, as well as the total bandwidth utilization of all elephants.

2.1 Latency-Based Throughput Calculation

We suspect that the third elephant’s sub-optimal link utilization is due to its high latency. The maximum throughput r can be expressed as a function of the advertised window and the latency:

$$r = \frac{window}{RTT}$$

The advertised window is $256pkts * 1500B/pkt$. Plugging this in yields

$$r = \frac{256 * 1500bytes}{844ms} = 3639kbps.$$

Since the latency is an underestimate given processing times and queuing delay, this supports the measured 3300 kbps that the last elephant achieves when transferring alone.

2.2 Improvement

To determine the optimal advertised window for this elephant, we use the inverse of the previous equation, plugging in the ideal 9.5 Mbps as the rate. Solving $9.5Mbps = \frac{window * 1500bytes}{844ms}$ for the window, we get an advertised window of 668 packets. The following plots show the cumulative bytes ACKed

and the bandwidth utilization of elephants after this improvement was made for the third elephant.

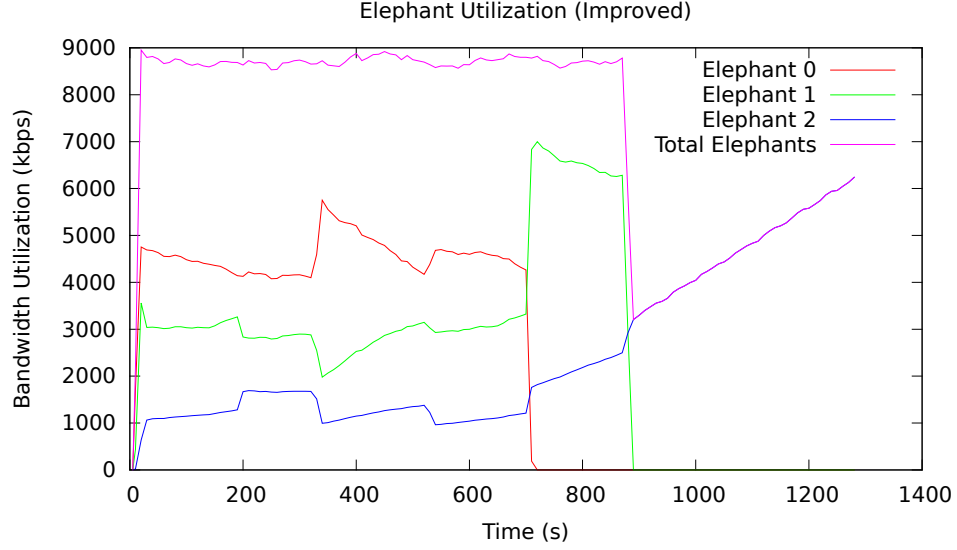


Figure 3: The bandwidth utilization of each elephant, as well as the total bandwidth utilization of all elephants.

As indicated in Figure 3, the third elephant experiences a ramp-up when left alone, and finishes earlier than before. The new elephant end times are 700, 875, and 1283 seconds, an improvement upon the previous 704, 885, and 1441 seconds. Figure 4 also shows the improvement as the slope of ACK evolution curves upward to reach nearly its previous slope after the second elephant finishes.

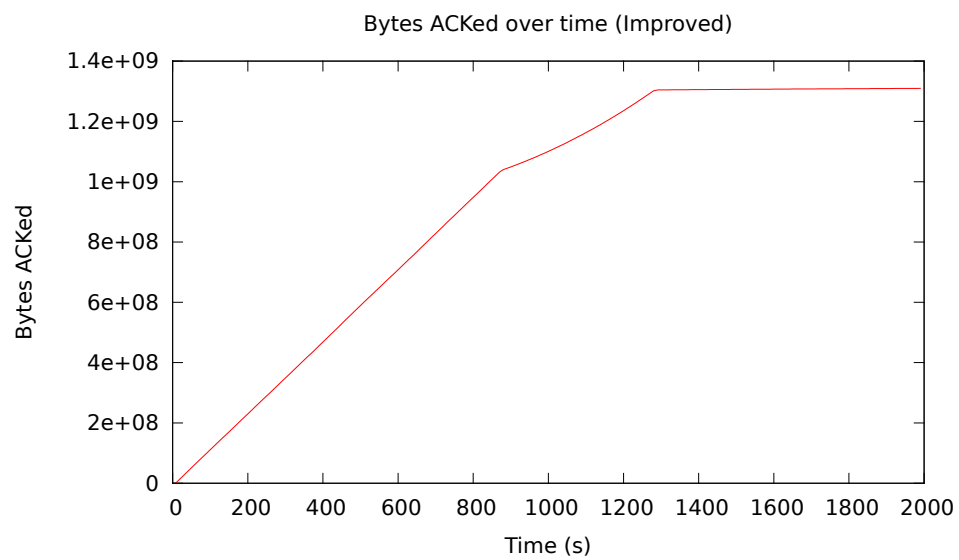


Figure 4: The cumulative number of bytes ACKed at the source, taken every 10 seconds.