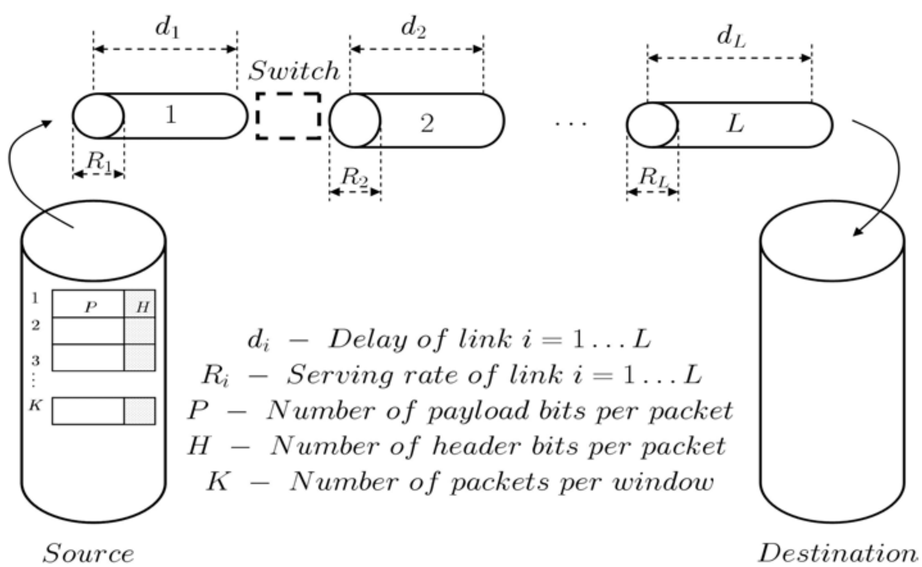
Bandwidth, Delay, Throughput and Some Math

By Oleg Berzin (Send feedback to feedback@ccieflyer.com)

Often the work in the field of network engineering and internetworking is not associated with exact science or precise computations (as for example in the field of mechanical engineering). The greatest evidence to that is the most popular answer a "seasoned" network engineer is likely to respond with when asked a technical question: "It depends…" Yes, the answer may depend on a number of variables and specific conditions that are hard to impossible to address with precise computations and simulated models. That is why, in my opinion, our profession depends heavily on the proof of concept testing, and that is why it is often considered as "more of an art" than "an exact science".

It is however also the case that internetworking and telecommunications in general are rooted in fundamental engineering principals. The terms that we use on a day to day basis (such as Bandwidth, Latency, Jitter, Throughput) have precise meanings and are defined in mathematical expressions. Sometimes the traditional meaning of the engineering term takes on a slightly different meaning in the internetworking. Take the term "Bandwidth" for example. For a radio engineer it would most likely mean the analog frequency range required to support a given signal. For a network engineer it means a serving rate or a speed of a communication link. The term "Throughput" is mostly associated with telecommunications but even in this field it may have different meanings, as in "the throughput of a link" or "a protocol throughput over a network path". We may not need to use math in our everyday tasks but sometimes it may make a difference.

Consider the following problem (see figure below). A batch of K packets needs to be transmitted from source to destination across a network path of L links with each link having a rate R_i bits per second and delay d_i seconds ($i = 1 \dots L$). Each packet has the same size with P bits of payload and H bits of protocol header. Ignoring the inter-link switching delays and assuming that packets are sent back-to-back by the source, compute the total delay to transfer the batch of K packets from source to destination.



Solution 1:
Approach: Compute the total delay of transmitting one packet from source to destination and then multiply by the number of packets.

For each link $i = 1 \dots L$: $\frac{P+H}{R_i}$ - is the packet serialization delay (the time to output $P + H$ bits onto a link of rate R_i bits per second), and d_i - is the propagation delay of link i. Therefore the total delay to transmit K packets:

$$D_1 = K\left(\sum_{i=1}^{L}\left[\frac{P+H}{R_i} + d_i\right]\right) \tag{1}$$

If all links had equal serving rates and propagation delays ($R_i = R,\ d_i = d, \forall i = 1 \dots L$), then:

$$D_1^e = KL\left(\frac{P+H}{R} + d\right) \tag{2}$$

Intuitively, Solution 1 makes sense. However it is incorrect because it implies that each packet has to wait until a previous packet reaches the destination. In other words, Solution 1 does not consider the "in-flight" packets.

Solution 2:
Approach: Compute the serialization delay for $(K-1)$ packets onto the first link and then add the serialization and propagation delays for the K_{th} packet.

$$D_2 = (K-1)\left[\frac{P+H}{R_1}\right] + \sum_{i=1}^{L}\left(\frac{P+H}{R_i} + d_i\right) \tag{3}$$

And, for equal links:

$$D_2^e = (K+L-1)\left[\frac{P+H}{R}\right] + dL \tag{4}$$

If we consider the difference between the two computations (for the case with all equal links):

$$D_1^e - D_2^e = (K-1)\left[(L-1)\frac{P+H}{R} + dL\right] \tag{5}$$

For example, if $K = 10,\ L = 5,\ P = 8,192\ bits,\ H = 320\ bits,\ R = 10,000,000\ bps$ and $d = 10\ milliseconds$, then:

$D_1^e = 542\ msec$, $D_2^e = 62\ msec$ and $D_1^e - D_2^e = 480\ msec$.

This simple example shows a very significant difference between the two methods of calculating the total delay. Specifically, the first method (however intuitive it may seem) leads to a huge mistake in calculations.

If we think about the Throughput it can simply be defined as a "number of bits divided by time needed to transport the bits". In case of a single link the link throughput (for the first link in the example) can be written as:

$$T_l = \frac{K(P+H)}{\left[K\frac{P+H}{R} + d\right]} \tag{6}$$

In case of the network throughput (the throughput of a network path in our example), it is:

$$T_n = \frac{K(P+H)}{D_2} \tag{7}$$

So for the example with all equal links:

$$T_l = \frac{10(8192+320)\ bits}{0.019\ sec} = 4.6\ Mbps$$

$$T_n = \frac{10(8192+320)\ bits}{0.062\ sec} = 1.37\ Mbps$$

At the protocol (e.g. TCP) level however, we have to factor in the time that it takes for the acknowledgement to get back to the source before the source can send another batch (window) of packets. Adding to the example the acknowledgement packet of size $A = 2048\ bits$, the time that it takes this packet to travel to the source is $D_a = L\left(\frac{A+H}{R} + d\right)$ and the protocol throughput (assuming window size of $KP\ bits$) can be written as:

$$T_p = \frac{K(P+H)}{(D_2 + D_a)} \tag{8}$$

Or, in the example above:

$$T_n = \frac{10(8192+320)\ bits}{(0.062+0.051)\ sec} = 1.32\ Mbps$$

Note that Throughput is not the same as the Link Serving Rate (or Bandwidth - for network engineers) and it always depends on delay. In fact achieving the Throughput that is equal to the link's Bandwidth would require that the propagation delay of that link be zero or nearly zero (see equation (6) with $d = 0$). However, if for example, we consider high bandwidth metro optical links – they do not have a zero propagation delay. Therefore, especially at a protocol, level a throughput that is equal to the link's bandwidth may not be achievable.

Another important note related to the protocol level throughput is that if the window size ($K \times P$ in equation (8)) is increased, and the propagation delay ($d \times L$ in equation (4)) remains constant – the throughput also increases. This increase, however, is limited by the maximum window size of a protocol. This effect is often referred to as the "Throughput-Delay Product": the product of throughput and delay is a constant ($T \times D = C$). Or in other words: in order to increase throughput, delay must be decreased: $T = \frac{C}{D}$.