Manually Testing Queueing Algorithms

Abiyaz Chowdhury

Rohit Chatterjee

Course instructor: Prof. Aruna Balasubramanian

I. Introduction

The entire point of the internet is enabling the transfer of packets from one end system to another. Being a huge network with a huge amount of effort going simply into the sustenance of the structure, the internet only provides best-effort delivery guarantees. It is no surprise, therefore, that a lot of thought goes into studying and dealing with packet lost. Packets may be lost in a variety of ways: by being directed to a wrong address, by not being correctly read or accepted and their end destinations, being corrupted or incorrectly split, due to congestion inside the network, and so on. Among these, congestion is often the most pervasive (and pernicious) type of packet loss, and so a great deal of effort goes into minimizing network congestion through various means. This is of course the motivation behind the celebrated congestion control mechanism in TCP, but there are other means to tackle this problem. Notably, TCP congestion control is implemented only at end systems, and therefore does not directly handle local congestion at midpoints or routers - which is really most of the network.

Recall that a router is a device that has multiple connections to other devices on the internet, and its basic function is to forward packets along correct paths to their destination. Thus router function is not immediate, since the router must determine where it must send each packet once it receives it. Further, each connection on the internet has its own capacity or bandwidth which fixes the maximum rate at which packets can be sent along this connection. The upshot of all this is that there is a certain maximum rate at which a given router can despatch incoming packets, and thus if it receives packets at a grater rate then there is congestion at the router. The way this is dealt with is straightforward: the router has a predefined amount of buffer memory to deal with stranded packets, and this is called the queue. The process of managing these packets and using the queue is called queueing. The most natural (and widely used) way to perform queueing is to do it in a first-in-first-out manner: the earliest recieved packet in the queue is sent out first by the router (hence the term queue), and the queue receives packets as long as packets come in faster than they can be sent out. Once the queue fills up, subsequent incoming packets are all dropped till the queue has space to accept more packets again.

On the face of things, one would not expect anything more from the queueing algorithm; there seems to be no apparent reason why the functioning of the queue should be related to the network performance. In other words, if the network is otherwise well-managed, we might expect such congestion to be short in duration, and therefore assume that the queue absorbs such bursty behaviour and in fact smoothens out traffic heading out from the router. However this is not always the case, and we can see that this method of queue operation may actually hamper network performance in some scenarios. For example,

II. Model Development

III. Model discussion

IV. Model evaluation

V. Conclusions

References

[1] James H. Stock, Mark W. Watson, *Forecasting inflation*, Journal of Monetary Economics 44 (1999) 293-335