Che-Chi Liu (Jack)

V00850558

CSC446 Project Report

Project: Priority Queueing to Alleviate Packet Re-ordering Problem

Introduction

The internet is like a courier that delivers data packets to the receiver (user). Just like courier services in real life, data packets being delivered might be lost, delay, or out of order. "In computer networking, out-of-order delivery is the delivery of data packets in a different order from which they were sent." One common way of applications to solve this problem is to use TCP as its transport protocol. Another solution is to "buffer out-of-order packets at the destination and re-order the packets into correct order before sending them to the application layer." The problem of this simulation project is to find out if re-ordering out of order data packets can be tone down with the help of priority queueing at the intermediate routers.

Simulation model

A simulation model for the problem can be implemented as a custom/server FIFO 2 queues system with prioritizing customer with younger ages over older ages, one high queue always serves younger ages first, then it serves older ages from low queue only if high queue is empty.

Simulation goals and parameters

The goal of this simulation is to alleviate re-ordering out data packets at destination.

First, the data packets need to travel from a source node to the router, then from the router to the destination. The simulation paraments are:

- Source node generates Poisson traffic, with average rate of 750 packets per second and 1125 packet per second. So, 1/750 and 1/1125 are inter arrival parameter with exponential distribution.
- Each packet has a fixed length of 1000 bytes and a unique sequence number.
- Source node's transmission speed is 10 Mbps. So, it takes 8000/(10^6)*10 = 0.0008 seconds for source to transmit a packet.
- From the source to the router, packets experience a random delay which follows a normal distribution with the mean of x seconds and the standard deviation of y seconds.
- From the router to the destination, packets experience a fixed delay of 50 ms = 0.05 seconds.
- Router's transmission speed is also 10 Mbps. So, it takes 8000/(10^6)*10 = 0.0008 seconds for router to transmit a packet.
- At router, there are 2 FIFO queues. One queue has higher priority than the other one. Each queue has a size of 10M bytes. Since each data packet has a fixed length of 1000 bytes, that means each queue can have 10,000 data packets.

Methodology

I built my simulation similar to a server/2 queue system. The data packets are the customer, the router is the server which provide service to transmit data packets to their destination.

The data packets coming in from source node as customer follows a Poisson distribution and with a random normal distribution delay. Therefore, I scheduled each data packet's inter arrival time as [exponential(1/lambda) + random delay + source transmission speed = 0.0008 sec] .

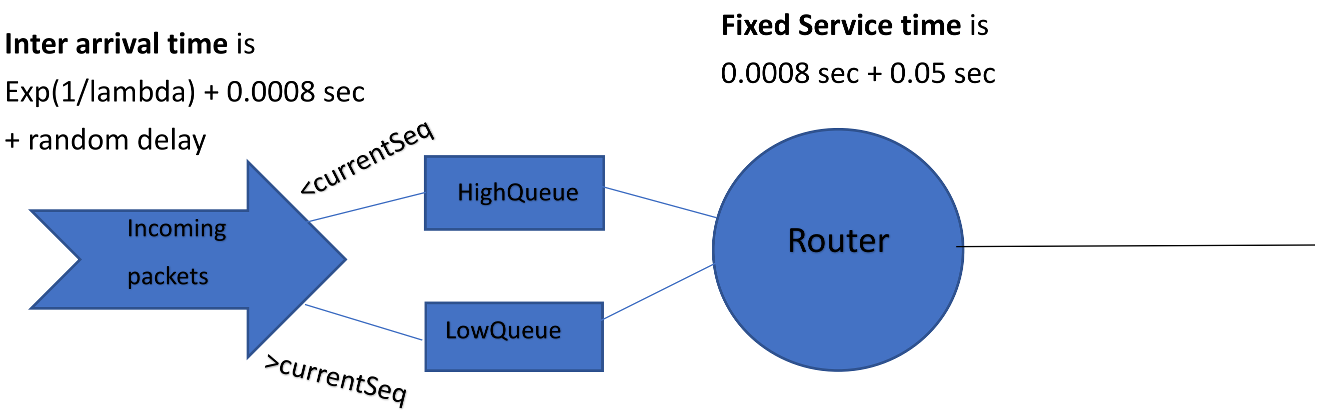
The router/server keeps track and updates of the largest sequence number it has seen so far, called currentSeq.

When data packet arrives at the router/server. If the router is idle, it sends the incoming data packet immediately. Else if the router is busy then it enters one of the 2 FIFO priority queues; if incoming data packet has a bigger sequence number than largest one that router has seen, it waits in the low queue. Else it waits in the high queue. Incoming packet will be dropped if their corresponding queue is full.

In networking, data packet arrival is random, we don't know which data packet will arrive at the router next. In other words, data packets coming in are random, we don't know which sequence number of data packets will arrive next at the router. Therefore, when I schedule arrival event for every data packet, I assign a unique sequence number randomly with the help of Collections.shuffle() method in java. After a shuffle, first element of a list of all possible sequence number is picked as the data packet's sequence number. Then the chosen sequence number is removed from the list.

Router always transmit data packets waiting in the high queue first because the goal of this simulation is to not to transmit out of order packets and decrease the number of re-orderings at destination. It sends the data packet with a lower sequence number(in order ones) than the one that the router has seen so far.

Each data packet's service time at router is the transmission speed from router to destination plus a fixed delay of 50ms (0.05 sec). Therefore, the service time for each packet is fixed.



For out of order rate, I count the number of times when incoming packet is bigger than currentSeq divided by total number of packets.

For average packet delay, I add each data packet's end to end delay divided by the total number of packets. End to end delay is the time for a packet to get from source to destination. End to end delay can be calculated by using the time when a data packets leaves the router after its service time minus its arrival time at the router.
If the data packet is sent immediately than the end to end delay can be calculated by adding the time to transfer from source node to router plus the time to transfer from router to destination, **0.0008 sec + random delay + 0.0008 sec + 0.05 sec.**

For packet loss rate, I count the number of times when incoming packet is dropped, since its corresponding queue is full, divide by total number of packets.

Analysis

750 packets per second, "server utilization" of 60%

X=0, Y=1	Packet out of order rate	Average packet delay	Average packet loss rate
1234 (Seed)	99.990%	0.8514490887668765	0
4567	99.981%	0.854651803444979	0
8911	99.985%	0.8510567925868902	0
22	99.984%	0.8519350981705394	0
6666	99.988%	0.8523327585071375	0
Average=	99.9856%	0.85228	

X=5, Y=10	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.989%	10.177105715943719	0
4567	99.987%	10.146817850701248	0
8911	99.986%	10.137471239695728	0
22	99.988%	10.147168618017088	0
6666	99.988%	10.160283266139469	0
Average=	99.9876%	10.15370	

X=20, Y=22.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.989%	27.505698649617964	0
4567	99.990%	27.48337897674444	0
8911	99.980%	27.58000441096177	0
22	99.985%	27.543337839655536	0
6666	99.989%	27.455468167233104	0
Average=	99.9866%	27.51357	

1125 packets per second, "server utilization" of 90%

X=0.9, Y=8.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.982%	7.171710849345768	0
4567	99.982%	7.173139250869115	0
8911	99.982%	7.160594244424613	0
22	99.982%	7.172246683632958	0
6666	99.985%	7.170981942944573	0
Average=	99.9826%	7.16973	

X=2.5, Y=4.2	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.983%	4.4989539022618416	0
4567	99.990%	4.482931972874236	0
8911	99.989%	4.49414243467136	0
22	99.984%	4.4951731309061253	0
6666	99.994%	4.4978591165295476	0
Average=	99.9880%	4.49381	

X=3.1, Y=5.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.995%	5.782707533424547	0
4567	99.989%	5.767600829625577	0
8911	99.994%	5.776949149316777	0
22	99.988%	5.785341121381726	0
6666	99.988%	5.794789738442377	0
Average=	99.9908%	5.78147	

Single FIFO Queue

750 packet per second, "server utilization" of 60%

X=0, Y=1	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.987%	0.8504062081882964	0
4567	99.988%	0.8535983956570677	0
8911	99.991%	0.8499843110004986	0
22	99.986%	0.8508928012004604	0
6666	99.987%	0.851271002088236	0

X=5, Y=10	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.983%	10.177036977017152	0
4567	99.989%	10.14675514880636	0
8911	99.989%	10.137407781131478	0
22	99.982%	10.096304277471628	0
6666	99.757%	10.160218447430196	0

X=20, Y=22.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.989%	27.50568172216336	0
4567	99.986%	27.48336521490561	0
8911	99.989%	27.579985975954034	0
22	99.989%	27.543313675013646	0
6666	99.988%	27.45545320729023	0

1125 packet per second, "server utilization" of 90%

X=0.9, Y=8.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.986%	7.1716023060496	0
4567	99.987%	7.173032497180753	0
8911	99.986%	7.160491103357977	0
22	99.990%	7.172142828198016	0
6666	99.983%	7.170875521029264	0

X=2.5, Y=4.2	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.986%	4.498823633446758	0
4567	99.986%	4.482802589402118	0
8911	99.986%	4.4939950734028884	0
22	99.995%	4.4950362786314596	0
6666	99.988%	4.497712192883717	0

X=3.1, Y=5.5	Packet out of order rate	Average packet delay	Average packet loss rate
1234	99.987%	5.782608569225942	0
4567	99.987%	5.767497804522031	0
8911	99.989%	5.776850808907576	0
22	99.990%	5.785232919177633	0
6666	99.983%	5.794674962268994	0

The simulation result above was computed with input of 100,000 data packets and a max queue size of 10,000. Comparing result from 2 FIFO priority queue and result with a single FIFO queue, the result performance metrices are really alike. The reason why the out of order rate is really high because it is possible for a big sequence number to get chosen from the list randomly for the first incoming packet, and the other incoming packets' sequence number will most likely to be smaller than the first big sequence number.

The average packet delay can also be computed without the source node and router transmission speed since their values are not significant enough to make a difference. End to end delay can be calculated just with the random normal distribution delay and the fixed 50ms (0.05 sec) delay. As you can see from the result table, each data packet's end to end delay is around the random delay time + 0.05ses. The random delay falls nicely under the bell curve between the mean value x and standard deviation y.

The reason why average loss rate is always equals to 0 is because most of the time the router is idle in this simulation, so it can transmit the packet immediately without taking up space of the queues.

Conclusion

Based on this simulation study, I conclude that using 2 FIFO priority queues at intermediate router does not really alleviate much packet re-ordering at destination. The intermediate router will always serve the next first packet waiting in high queue first before serving packets waiting in low queue. **Due the randomness of normal distribution delay and randomness of packet arrival, out of order packets still exist inside the high queue**. The “out of order” packets go into the high queue simply because their sequence numbers are smaller than the one router has seen but they are still out of order inside the high queue. Since out of order packets still exist inside the high queue and the high priority queues is FIFO, packets are still getting transmitted to destination out of order. Using 2 FIFO priority queues at intermediate router is almost equivalence with using single FIFO queue.