Assignment 2

CS544 : Topic in Networks

REPORT TITLE

Implementations of different scheduling Algorithms in a Packet Switch

Prepared By :

Avadhesh Sharma (194101011)
Arunav Saikia (194161001)

Indian Institute of Technology, Guwahati

Contents :

Abstract

In packet scheduling algorithm of a switch has an arbiter with every port (including both Input and Output Ports). Input Port arbiter helps sequence of packets to be selected from input queues while Output Port arbiter manages packet sequence to be transmitted from output queues.
Packet Scheduling algorithms can be referred as Network Scheduler.
Queuing system is an integral part of Network schedular where buffer is usually reserved for packets residing both side of the switch.
Selection of packets from Input queue and Output queue depends on scheduling mechanism.
Sometimes for some scheduling mechanism, it is not feasible to schedule all the generated packets given the requirements of the system so here comes the decision where our scheduling mechanism has to decide which packet to forward and which one to drop.

Introduction

The objective of this assignment is to measure the performance of queuing in the packet switch.
We measure the performance by comparing the Average Link Utilization and Average Packet Delay for different scheduling algorithms for given Parameters of the switch.
In the assigmment, we will demonstrate three Packet Scheduling algorithms named as INQ, KOUQ and ISLIP.
The below three phases are implemented by above mentioned algorithms -

- Traffic Generation
- Packet Scheduling
- Packet Transmission

All above three phases will take place at the beginning of each time slot.
We are generating the graphs for the performance measurement with the help of Output data generated till a reasonable amount of time (Max Time Slot)

Implementation

This section of the report gives detailed explanation of the implementation of the scheduing algorithm stated in the Introduction section.

Input to Program:

The following format has to be used to run the program -

./Scheduler −N switchportcount −B buffersize −p packetgenprob −queue INQ | KOUQ | ISLIP −K knockout −out outputfile T maxtimeslots

Example (containing default values) :
./Scheduler −N 8 −B 4 −p 0.5 −queue INQ −K 4 −out outputfile T 10000

- Number of switch input port and output port : We have N input ports and N output ports. Default value of N is 8.
- Buffer Size : Default buffersize is 4. Each port can have up to B fixed length packets.
- Packet generation probability : Default probability is 0.5. A packet is generated with this probability from each port in a given slot.
- Queue scheduling technique being used : -queue argument specifies the queueType for the Program. Default queueType is “INQ”.
- MaxTimeSlot : This value depicts the simulation time. Default value of simulation time is 10000.
- Outputfile : -out argument is used for a file to have output information.

For all algorithms, all packets are of same length. Time is slotted and one time slot is equal to the transmission time of one packet.

Outputs

Below are the performance metrices which are to be measured:

- Average Packet Delay : It is calculated with the help of summation of packet delay of each packet divided by all transmitted packet.

- Average Standard Deviation : It is calculated with the help of summation of standard deviation of each packet divided by all transmitted packet.

- Average Link Utilization : First we calculate link utilization of each link which is defined as the fraction of time has been used for transmitting a packet in the entire simulation time. Average link utilization is the mean value of all individual link utilization.

- KOUQ Drop Probability : This probability per slot comes into picture when more than K packets are generated for an output port. For e.g, if more than K packets are generated for 3 out of 8 output ports in a given slot then the probability for that slot is 3/8.
  We have calculate average drop probability over all the slots for the simulation duration.

N	p	Queue type	Avg PD	Std Dev of PD	Avg link utilization

The outputfile has the above mentioned format for the output after program terminates.

Phases in the Algorithm

- Traffic Generation: Each input port will generate a packet with the given probability “packetgenprob” in this phase. Output port for each packet is determined randomly with the help of uniform probability from the set of all output ports. The start time of each packet is randomly set between t+0.001 to t+0.01 where “t” is the current time slot.
  We can ignore offset value in arrival time while calculating packet delay but this offset value is useful when we decide which packets to forward in the case of “KOUQ” scheduling algorithm.

- Scheduling: In this phase, we schedule packets generated in previous phase.

  1. INQ : This is the simple scheduling algorithm where if there is no contention for an output port for a generated packet in a time slot

then this packet is selected for transmission and placed in the corresponding output port's queue buffer.
We can have many packets contending for the same output port in a time slot, we will select one packet among them randomly and will place it in corresponding output queue's buffer and all remaining packets will remain in the input queue.
In INQ, we have same buffer size for both sides.

2. KOUQ : In this algorithm, Output buffer size is "knockout" value provided in command line arguments and is also referred as "K".
Here a maximum of K packets are queued (based on packet arrival time) at the corresponding output port.
Packets can be queued in any order if they have same arrival time.
If more than K packets are there for an output port in a given slot "t" then K packets are selected randomly for buffering and remaining packets get dropped.

3. ISLIP : [1] paper expains in detail how ISLIP works.
This paper compares similar algorithms like Round – Robin Algorithm and PIM (Parallel Iterative Matching) with ISLIP and justifies the reasons why ISLIP performs better than them.
We will explain the algorithm in simple terms.
Here one timeslot is similar to one Round and one round can have multiple iterations (till we have maximum size matching from input port to output port).

Below three steps happen in an iteration of the algorithm:

(1) Request: In this step, all unmatched input ports ask desired output ports if they can send the packet

(2) Grant: Here each unmatched outport port has to determine which input port can send data using all received requests. If there is only one received request then that request will be entertained.
Round – Robin method will be used to select from received requests if there are multiple received requests for an output port.

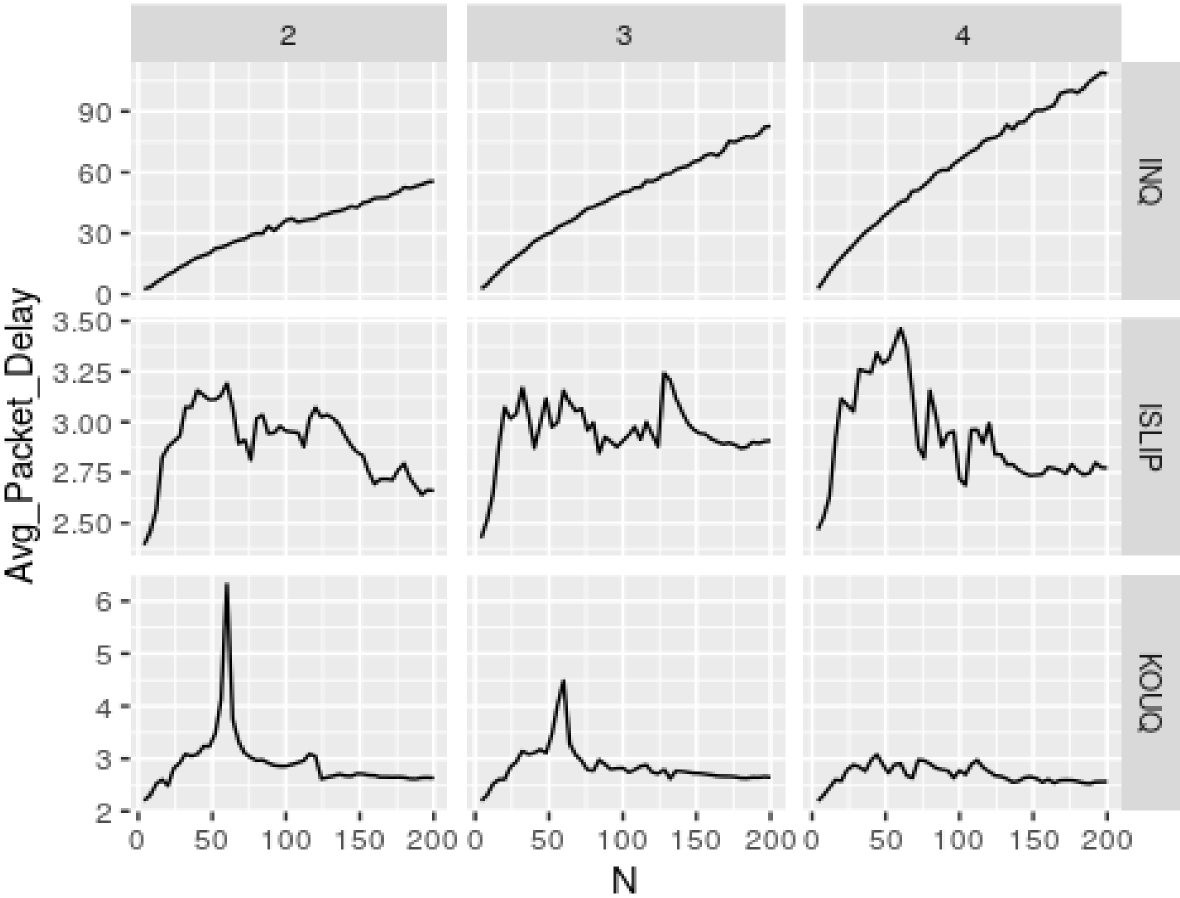
Now the output port sends a “Grant” message to the selected input port stating that you can send the data now.

(3) Accept : An unmatched input port collects all the grant messages received from output ports and selects one in Round-Robin fashion. After accepting the grant, input put notifies output port that its grant has been accepted.

After a round or mulitple iteration or a time slot, a maximum matching size is found out and number of input-output matches are increased which means that we can send more packets through the crossbar at a time slot.

Performance Measurement among all algorithms shown with the help of Graph :

1. Average Packet of INQ, KOUQ and ISLIP based on their Buffer Size :



Here, we have used a facet-grid to analyse the performance measure of INQ, KOUQ and ISLIP. Here, we have split the plot into facets, subplots that each displays one subset of the data. The columns of the facet categorize the BufferSize of the Switch and the rows categorize the Queue type.

INQ:
From the above graph as we have seen that, in all the three cases(i.e. for BufferSize = 2,3and 4) the Average Packet Delay increases with the increase in no of ports. However, the slope of the curve increases as the buffer size increases.

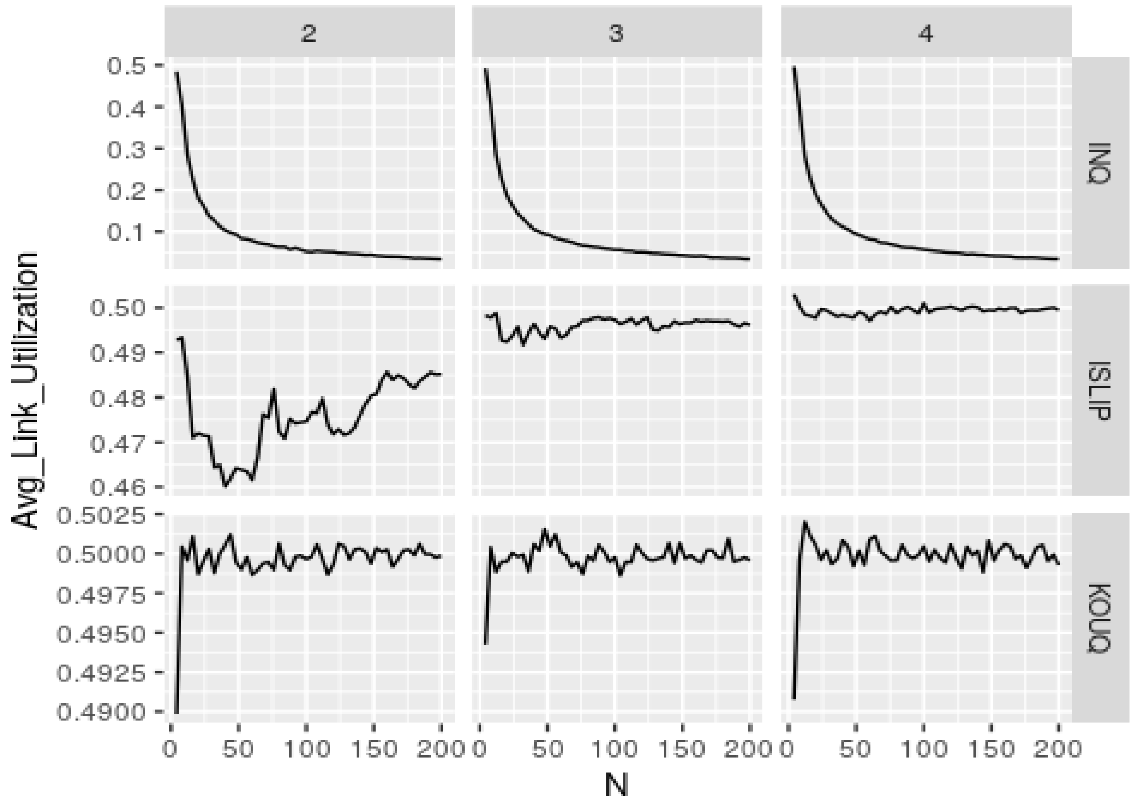
In INQ, Packets can be queued in input buffer when more packets are contending for a desired output port so the response time for packets may increase.

ISLIP:
In ISLIP, for all the three cases Average Packet Delay start from a lower value less then 2.5 but it gradually increases to maximum average delay of 3.50 for Buffer Size of 4.Here the average packet delay fluctuates between 2.50 and 3.50. ISLIP in all the three cases stabilizes after a certain number of Ports.
In ISLIP also, we use Virtual Output Queue for handling the contention. Here maximum matching size is found out in every round so the average packet value ranges between INQ and KOUQ.

KOUQ:
Here K values is 0.6*N where N is the total no of ports.
As seen in ISLIP the same trend is seen here for the starting values of N. But for Buffer Size of 2 average packet delay increases upto a value greater than 6. For Buffer Size of 3 average packet delay increases upto a value of 4.5. For Buffer Size of 4 average packet delay increases upto a value greater than 3. After reaching the maximum point the average packet delay declines to a value less then 3 for all the three cases and the it fluctuates between a range of 2.5 to 2.75.

From the above analysis we have found that ISLIP perform better then KOUQ and INQ for Buffer Size of 2,3 and 4. As the average packet delay range between 2.50 to 3.50. For the INQ, average packet delay increase as we increase the number of ports. For KOUQ, average packet delay range between 2 to 6.Therefore, KOUQ is better then INQ but not better then ISLIP.

2. <u>Average Link Utilization of INQ, KOUQ and ISLIP based on their Buffer Size:</u>



INQ:
Here, as the number of ports increases, the average link utilization decreases exponentially and after a certain number of ports the average link utilization declines at a stable rate. Here, all the three graphs show similar characteristics for different size of Buffer.

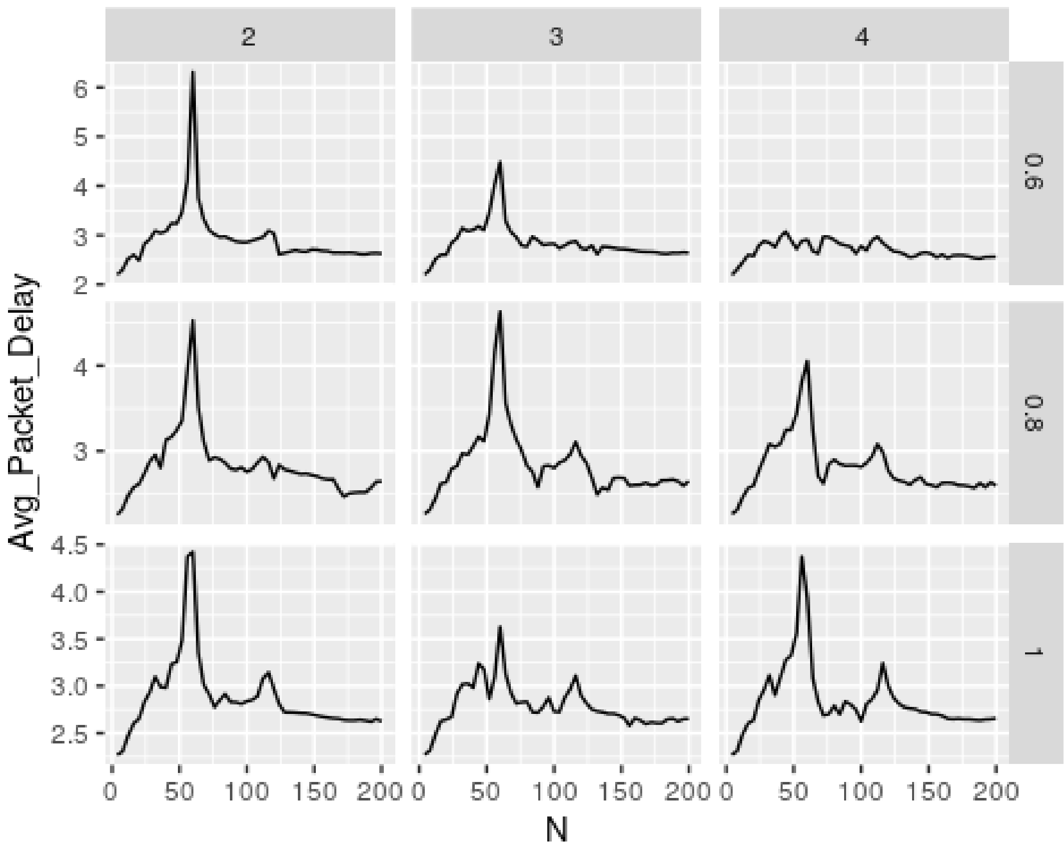
ISLIP:
For Buffer of Size 2 the average link utilization graph shows a fluctuating behaviour. It decreases to a minimum value of 0.46 and the then increases to become stable as we increase the number of ports.
For Buffer of Size 3 the average link utilization fluctuates between a range of 0.49 to 0.50.

For Buffer of Size 4 the average link utilization stabilizes as the number of ports are increased to a value slightly above 0.50.

KOUQ:
For all the three size of Buffer the average link utilization starts with a lower value and then around 0.5000.

As we can see that if no of ports in crossbar or switch are less than ISLIP link utilization performance is better than other two and as no of ports increases, both ISLIP and KOUQ perform same because if N is larger then K value (0.6 * N) is also higher so there is very less chance for a generated packet to get dropped.

3. Average Packet delay of KOUQ for different values of K based on Buffer size :



In the above facet-grid, we analyse average packet delay of KOUQ Queue with different values of k (i.e. k = 0.6*N,0.8*N and 1*N) and differnet buffer size.

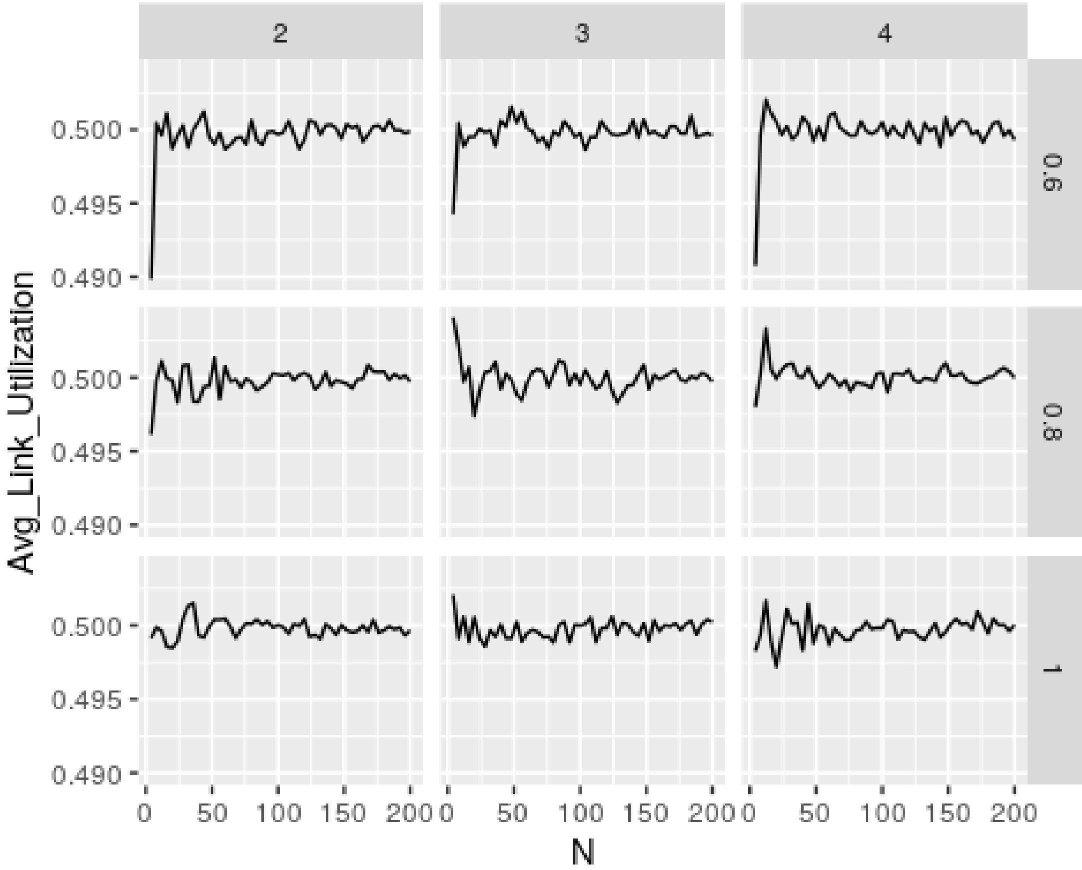
For K = 0.6*N
For Buffer Size of 2 average packet delay increases upto a value greater than 6. For Buffer Size of 3 average packet delay increases upto a value of 4.5. For Buffer Size of 4 average packet delay increases upto a value greater than 3. After reaching the maximum point the average packet delay declines to a value less then 3 for all the three cases and the it fluctuates between a range of 2.5 to 2.75.

For K= 0.8*N
All the three graph show similar trend. In all the three case the average packet delay starts with a lower value and then increases to a maximum value less then 5 and then the graph becomes stable as the number of ports are increased.

For K = 1*N
All the three graph show similar trend as shown for K= 0.8*N. In all the three case the average packet delay starts with a lower value and then increases to a maximum value less then 5 and then the graph becomes stable as the number of ports are increased. For Buffer of size 2, average packet delay increases to a maximum value less then 3.75 and then becomes stable as the number of ports increases.

If we increase the K value then Output buffer size gets increased due to which KOUQ drop probability decreases which increases the response time of packets showing the increase in average packet delay.

4. Average Link Utilization Values of K Based on Buffer Size :



In the above facet-grid, we analyse average link utilization of KOUQ Queue with different values of k (i.e. k = 0.6*N,0.8*N and 1*N) and differnet buffer size.

For K = 0.6 * N
For all the three different values of Buffer Size, average link utilization starts with a lower value of 0.490 except for Buffer Size of 3 which starts with a higher value of 0.495. In all the three subplots as the number of ports increases the graph keep on fluctuating between the values 0.495 to 0.505.

For K = 0.8 * N
For all the three different values of Buffer Size, average link utilization starts with a much higher value of 0.496. In all the three subplots as the

number of ports increases the graph keep on fluctuating between the values 0.495 to 0.505

For K = 1*N
We can see that as K increases , average link utilization is same for higher values of N for any K. As we can see that when N is very less, k = 1*N gives better link utilization because it has less chance for the packets to get dropped as compared to k = 0.6*N, k = 0.8*N because k=1*N has more buffer buffer size .

Average KOUQ drop Probability:

As K increases , the average KOUQ drop probability decreases which can be seen from the increasing values of K stated below -

N = 64 , K = 1 Average KOUQ drop Probability = 0.114747
N = 64, K = 5 Average KOUQ drop Probability = 0.005259
N = 64, K= 10 Average KOUQ drop Probability = 0.0002484.

Programming Language Used and Tools Used:

Visual Studio Code, Rstudio (for Graphs only) , C++, R (for Graphs only)

Conclusion:

It can be seen from the above analysis of the performace metrices that ISLIP performs better than other two in terms of Average link utilization and Average packet delay as no of port increases.
When we talk about KOUQ, we can state that as the output buffer (K) increases, KOUQ performance is up to the mark and Average KOUQ drop probability decreases.

References

[1]. The iSLIP Scheduling Algorithm for Input-Queued Switches Nick McKeown, Senior Member, IEEE