Assignment 2

CS 544: Topics in Networks
Deadline: 11:59 pm, 20 April 2020

Things to note before you start:

- No extensions in submission are allowed. Delay in submission will lead to penalty in marks.
- The programs can be written in C/C++/Java.
- Assignments submitted before the deadline will only be considered for evaluation.
- Submissions should be done via Moodle. You can access it via https://www.iitg.ac.in/moodle. Please do not email your assignments separately to the TAs, it will not be considered for evaluation.
- Please read the question carefully.
- Your code will be checked for plagiarism. Any kind of academic dishonesty, plagiarism, etc. will lead to penalties.
- NO sharing of code between students, submission of downloaded code (from the Internet, Campus LAN, or anywhere else) is allowed. The first instance of code copying will result in ZERO marks for the assignment. The second instance of code copying will result in a 'F' grade. Students may also be reported to the Students Disciplinary Committee, which can impose additional penalties.
- Please protect your Moodle account password. Do not share it with **ANYONE**, including your teammate. Do not share your academic disk drive space on Campus LAN.

1. The purpose of this assignment is to understand the performance of queuing in a packet switch. Your program should implement different types of scheduling mechanisms, as described below.

Inputs

The command line will specify the following:

- Number of switch input and output ports

- Buffer size
- Packet generation probability
- queue scheduling technique being used

For e.g., the input can be given as below:

$./routing\ -N\ switchportcount\ -B\ buffersize\ -p\ packetgenprob\ -queue\ INQ|\ KOUQ|\ ISLIP\ -K\ knockout\ -out\ -outputfile\ -T\ maxtimeslots$

- The switch is assumed to have N input and output ports. Ports are numbered from 0 through $N-1$. You can consider the default value of N as 8.
- All packets are of same length. TIme is slotted. One time slot equals the transmission time of one packet.
- Default buffersize B is 4, i.e, each port will hold upto B fixed length packets.
- Default *packetgenprob* is 0.5. It denotes the probability that an input port will generate a packet in a given slot.
- The $-queue$ argument specifies the queue type; the output file will contain the output generated by the program. Default is : INQ.
- Default value of *maxtimeslots* is 10000. It specifies the simulation time.

Switch Operation

The program will consist of three phases:

- Phase 1 – Corresponds to traffic generation.
- Phase 2 – Corresponds to packet scheduling.
- Ohase 3 – Corresponds to packet transmission.

All three phases will take place at the beginning of each time slot. Initially, all the packet queues are assumed to be empty.

Traffic Generation

In this phase, each port will generate a packet with probability *packetgenprob*. A packet's destination port is selected randomly with uniform probability from the set of all output ports. The start time of each packet is randomly set between $t + 0.001$ and $t + 0.01$. For delay calculation purposes, ignore this offset and assume that the arrival time is t.
Packet delay is calculated as the difference between transmission completion time and the packet arrival time.

Scheduling

In this phase, the packets generated in the previous phase are handled.

- **INQ:**– For each packet generated, if there is no contention for its desired output port, it is selected for transmission and placed in the corresponding output port's buffer. For packets contending for the same output port, one of the packets is randomly selected for transmission and placed in the corresponding output port's buffer; the other packets are queued at the corresponding input port.
- **KOUQ:**– A maximum of K packets (per output port) that arrive in a given slot are queued (based on packet arrival time) at the corresponding output port. If two or more packets have the same arrival time, the packets can be queued in any order. If more than K packets arrive in a slot for a particular output port, then K packets are randomly selected for buffering, and the remaining packets are dropped. The default value is $K = 0.6N$.
- **iSLIP:**– Implement the iSLIP scheduling algorithm with a Virtual Output Queue (VOQ). Follow the reference on iSLIP scheduling algorithm that was posted earlier on Moodle.

Transmission

At each output port, the packet at the head of the queue is transmitted. In case of KOUQ, the packet at the head of the queue will have lowest arrival time.

Outputs

The performance metrics to be measured:

- Average packet delay: The mean packet delay computed for all transmitted packets.
- Average link utilization: Link utilization for each link, is defined as the fraction of time a link has been used for transmitting a packet, with respect to the entire simulation duration. Average link utilization would be its mean value.
- KOUQ drop probability: The probability per slot that more than K packets were generated for an output port. For e.g., if in a given slot, more than K packets were generated for 3 out of 8 output ports, then the probability for that slot is 0.375. You should report the average probability over all the slots for the simulation duration.

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

Table 1:

The program, upon termination, will give the following line as output (may be separated by tabs), that will be appended to the specified output file.

A technical report that compares the performance of the scheduling schemes has to be submitted. The report should consist of performance graphs comparing average packet delay and average link utilization of different values of N, $L \in 2, 3, 4$ and $K \in 0.6, 0.8, 1.0 \times N$. N can be varied from 4 to as high as possible, while ensuring that the simulator code terminates in a reasonable amount of time. Since all the communications, corrections, and evaluations are going to happen only via online mode, the quality of your report becomes very important. Easily viewable graphs, detailed technical explanations, conclusions, and writing of the report (including, grammar and typos) are of utmost importance.

Do upload the assignment in .zip format including source code files, sample output files, technical report, README, and comments files. Your code should be easily readable for evaluation and it should be very clear as to how to run the program and how to generate the output.
Name the .zip/tar.gz file you are submitting the format - "AssignmentNo.–Roll no. of Group members separated by underscore".

Grading

- INQ: 10 points
- KOUQ: 20 points
- iSLIP: 35 points
- Report: 25 points
- Comments, README: 10 points