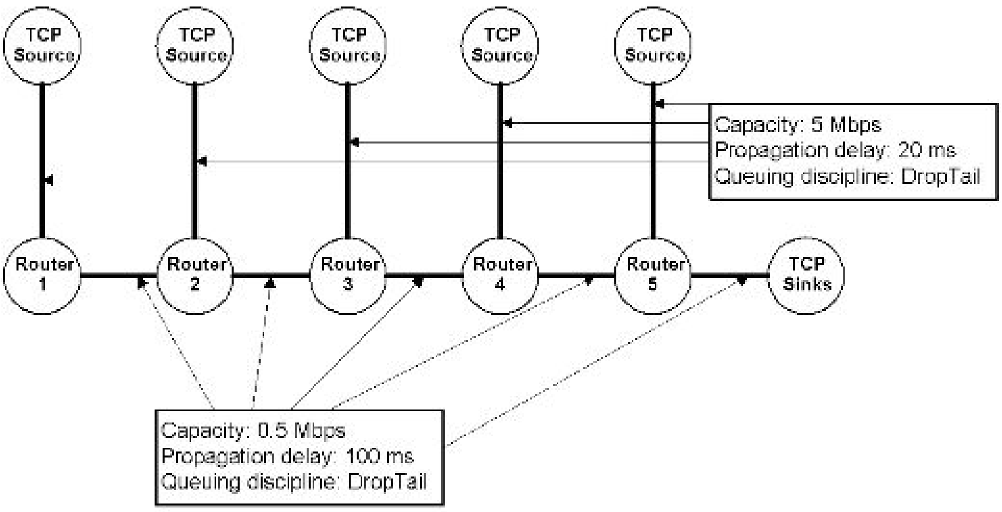
**Name: Thiar Hasbiya Ditanaya**
**NRP: 5116201048**

Programming Assignment 2 (with NS-2 simulator) Analysis

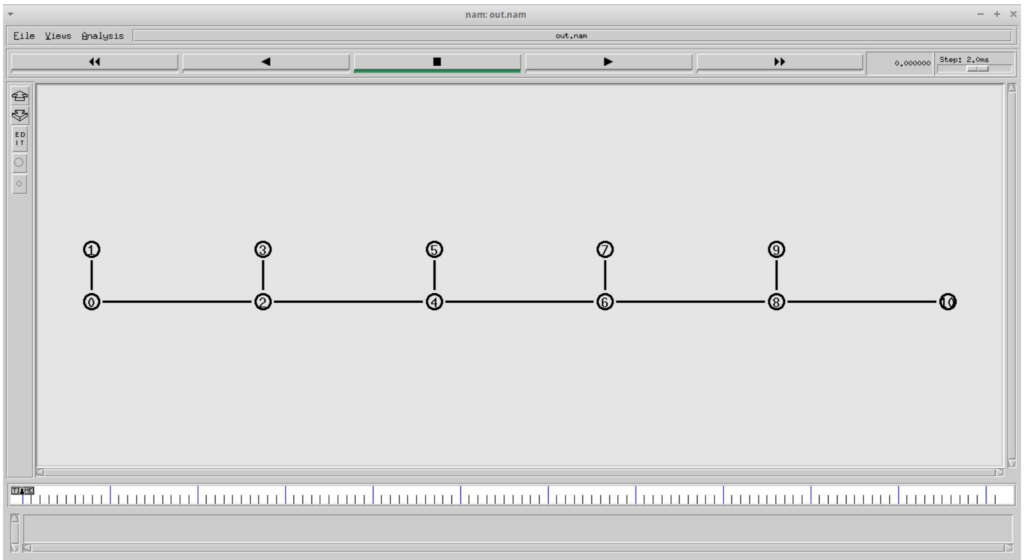
In this assignment we must design network topology described below. We applied various network queue algorithm like droptail and sfq. Also we encourages to try another algorithm like RED,DDR,FQ. The simulation using NS2 , nam and xgraph to create scenarios , run the simulation and draw the graph according the data generated by scenarios. After that we must analyze the network performance using throughput in each node. To analyze the throughput we use graph that drew by xgraph.



The network topology

First we must create network scenarios using ns2, source code packed in another file.

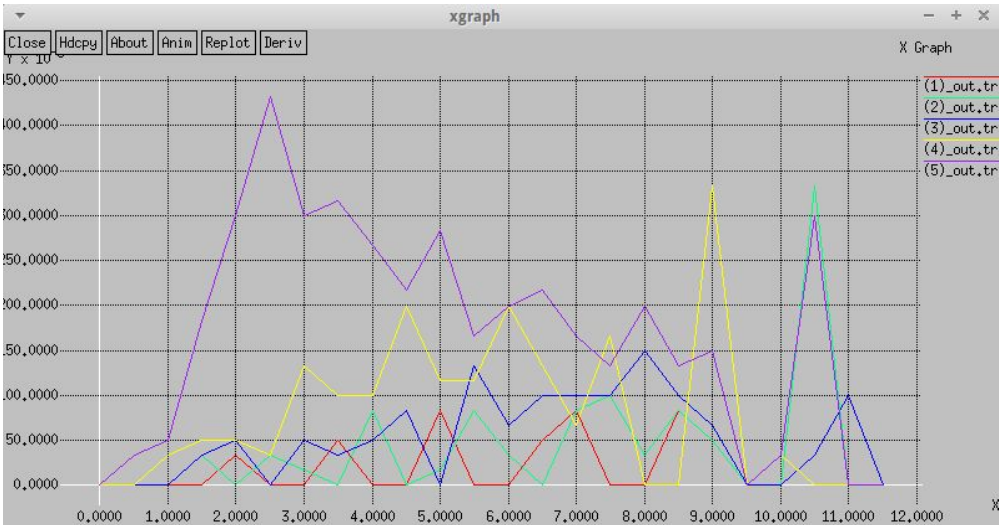
After we run the .tcl file using ns command like **ns drop-tail.tcl**, there will be files out.nam,all_out.tr, 1_out.tr, 2_out.tr, 3_out.tr, 4_out.tr and 5_out.tr . Simply run file out.nam using command **nam out.nam**. Then there will be simulation like we described in scenarios. Then we can play the simulation using play button.



The files 1_out.tr, 2_out.tr, 3_out.tr, 4_out.tr and 5_out.tr we use to draw graph using xgraph. Then we can analyze the performance of each queue algorithm.

- Droptail

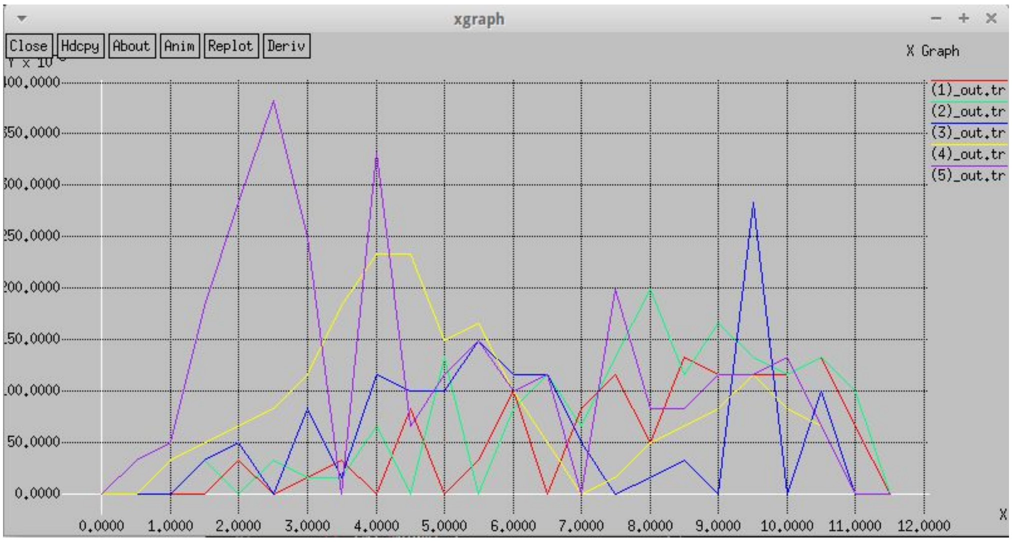
    Tail drop or Drop tail is simple queing algorithm used by router to decide where packet to drop. Drop tail will drop the last packet enqueued to the queue if the queue full. According to experiment we can draw graph below



The result show us, that the throughput on each node isn't fair enough. The node nearest to the sink will have more throughput than the others.

- SFQ

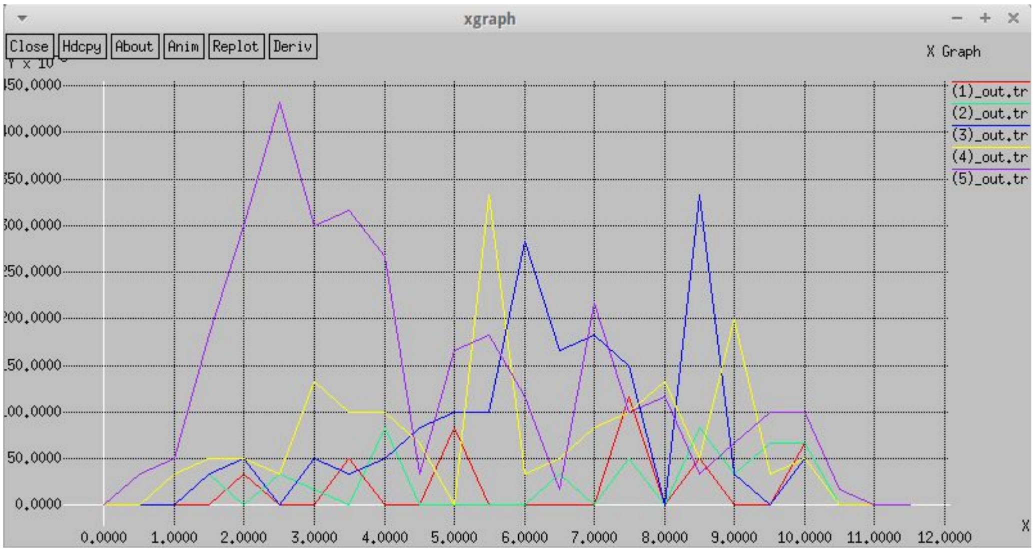
    SFQ or **Stochastic fairness queuing** is one of queue algorithm. SFQ does not shape traffic but only schedules the transmission of packets, based on 'flows'. The goal is to ensure fairness so that each flow is able to send data in turn, thus preventing any single flow from drowning out the rest.
According to experiment we can draw graph below



The result show us, that the throughput on each node is fair enough than the DropTail algorithm.

- RED

RED or **Random early detection** is one of queue algorithm. RED monitors the average queue size and drops (or marks when used in conjunction with ECN) packets based on statistical probabilities. If the buffer is almost empty, all incoming packets are accepted. As the queue grows, the probability for dropping an incoming packet grows too. When the buffer is full, the probability has reached 1 and all incoming packets are dropped. According to experiment we can draw graph below

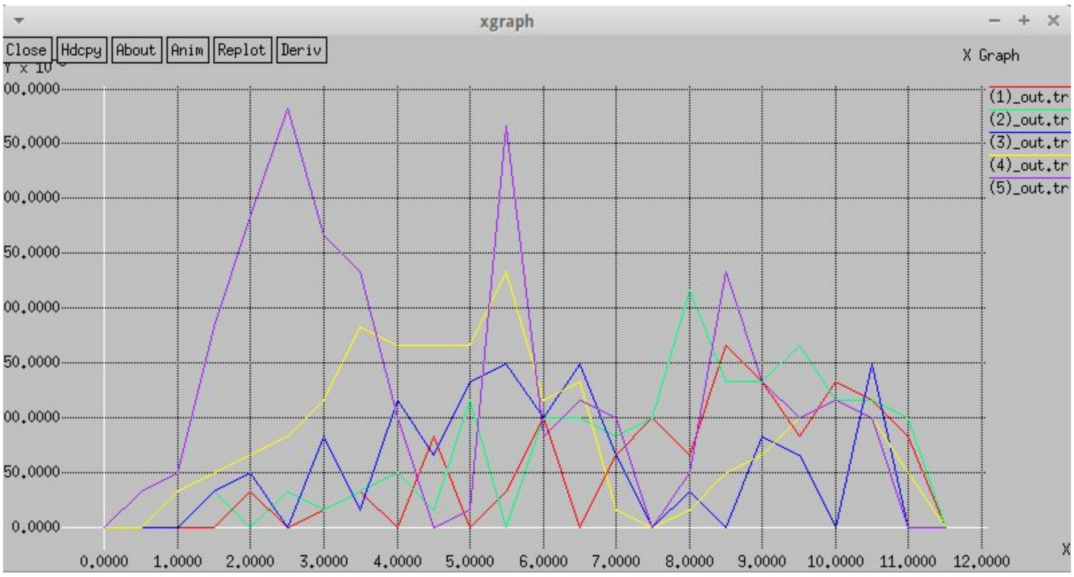


The result show us, that the throughput on each node is fair enough than the DropTail algorithm.

- DDR

DDR or **Deficit Round Robin** (**DRR**) is a scheduling algorithm for the network scheduler. DRR is, like weighted fair queuing (WFQ), a packet-based implementation of the ideal Generalized Processor Sharing (GPS) policy. It was proposed by M. Shreedhar and G. Varghese in 1995 as an efficient (with *O(1)* complexity) and fair algorithm.

According to experiment we can draw graph below

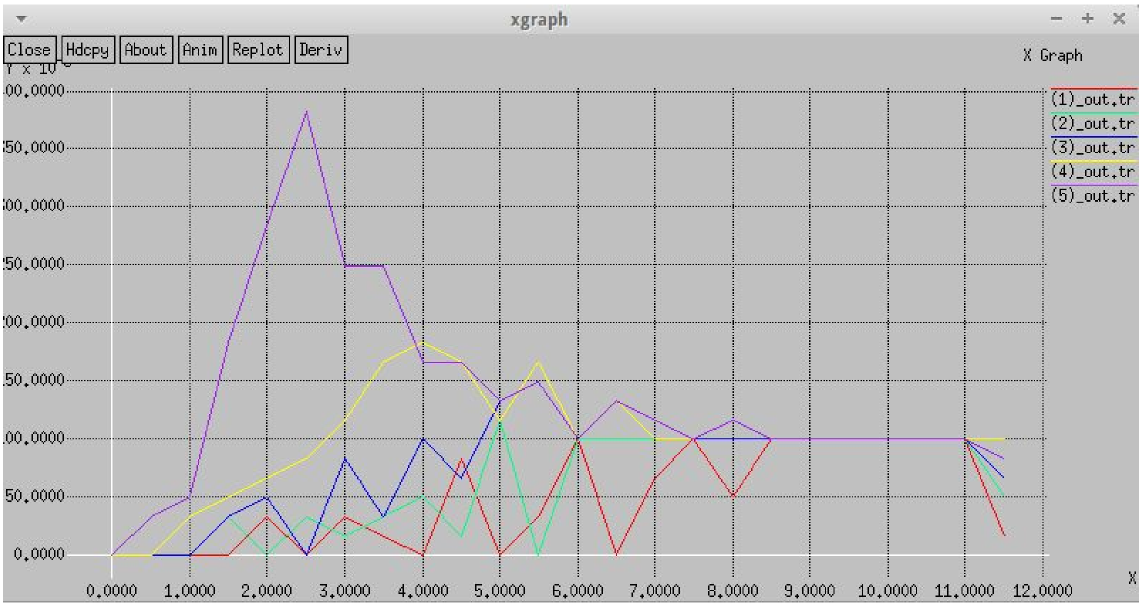


The result show us, that the throughput on each node is fair enough than the DropTail algorithm.

- FQ

FQ or **Fair queuing** is a family of scheduling algorithms used in some process and network schedulers. The concept implies a separate data packet queue (or job queue) for each traffic flow (or for each program process) as opposed to the traditional approach with one FIFO queue for all packet flows (or for all process jobs). The purpose is to achieve fairness when a limited resource is shared, for example to avoid that flows with large packets (or processes that generate small jobs) achieve more throughput (or CPU time) than other flows (or processes).

According to experiment we can draw graph below



The result show us, in 9.0 seconds this algorithm make the most fair decision among all others algorithm.

## Conclusion

The experiment show us, the different algorithm for queuing can make better fairness throughput to each nodes. The FQ algorithm is the most fair algorithm for dropping packets.

## Source

http://nile.wpi.edu/NS/simple_ns.html
http://www.isi.edu/nsnam/ns/tutorial/nsscript4.html
https://en.wikipedia.org/wiki/Network_scheduler
https://en.wikipedia.org/wiki/Tail_drop
https://en.wikipedia.org/wiki/Random_early_detection
https://en.wikipedia.org/wiki/Deficit_round_robin
https://linux.die.net/man/8/tc-sfq
https://en.wikipedia.org/wiki/Fair_queuing