**Project Report on Statistical Comparison Between Various Queue Disciplines of Active Queue Management Algorithm of Network Scheduler in TCP Internet Router**

by

**Tanmay Rajpurohit**

**(GT ID# 901563229)**

**Chandra Khatri**

**(GT ID#902988349)**



Submitted In Partial Fulfillment of the Requirements for the

Course CSE 6730: Modeling and Simulation

Course Instructor

**Prof. Richard Fujimoto**

Georgia Institute of Technology

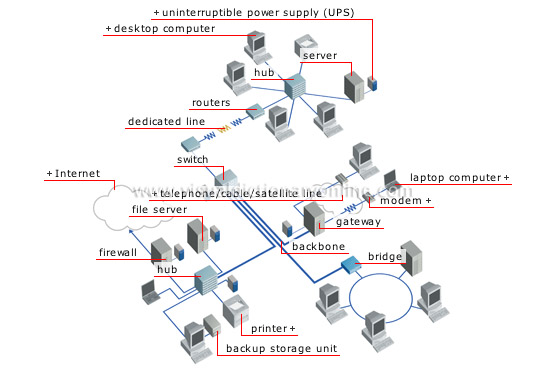
March 2015

**Problem Description**

The system under investigation (SUI) is computer network, static topology, communicating with each other through packet switching mechanism employing virtual circuit algorithm or datagram using TCP/IP protocols.

The endpoints or leafs of the network are computers which are source and sink of data packet that travel through the network. The nodes are routers, switches, bridges, gateways etc depending upon the network type LAN/WAN etc and uses the algorithm of store and forward using first hop routing methodology.

An Internet router typically maintains a set of queues, one per interface, that hold packets scheduled to go out on that interface. Historically, such queues use a *d* *rop­tail* discipline: a packet is put onto the queue if the queue is shorter than its maximum size (measured in packets or in bytes), and dropped otherwise.



The buffer works as a queuing system, storing the network packets temporarily until they are transmitted. The network scheduler logic decides, in a way similar to statistical multiplexers, which network packet to forward next from the buffer. On a node in packet switching communication network, a network scheduler, also called packet scheduler, is an arbiter program that manages the sequence of network packets in the transmit and receive queues of the network interface controller, which is a circular data buffer.

Due to Denial of Service (DoS) attacks or high utilization the classical FIFO with drop tail discipline are prone to TCP congestion collapse, and therefore Active Queue Management Algorithms came to aid. Active queue disciplines drop or mark packets before the queue is full. Typically, they operate by maintaining one or more drop/mark probabilities, and probabilistically dropping or marking packets even when the queue is short.

The goal of the project is to address and analyze the following questions

1. For a given parameter inputs **router throughput** and **buffer size** compare the network efficiency of various Active Queue Management Algorithms (Random Early Detection, Blue and Stochastic Fair Blue, Advanced Random Early Detection etc.) as against the classical FIFO with tail drop through Discrete Event Simulation by analyzing the primary metrics of interest (**latency**, **average buffer occupancy**, **average packet drop/ retransmission)** that eventually preventing TCP congestion collapse.
2. Design and implement the same simulation in parallel simulations scheme using deadlock detection and avoidance using Chandy-Misra-Brayant algorithm.

**Literature Review**

1. L. G. Birta and G. Arbez, Modeling and Simulation: Exploring Dynamic System Behavior, Springer, 2007.

**Conceptual Model**

**Constants and Parameters**

1. Value of buffer size and throughput.
2. Stochastically generated but fixed for each simulation run the average data generation rate **λ’s** for each computer.

**1.Inputs**:

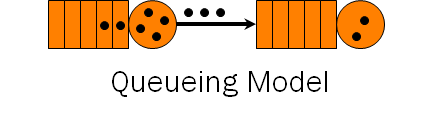
1. Exponential distribution of data packets generated form the average packet generation rate **λ’s** for each computer,
2. Transmission look ahead time of data travel between the two nodes,
3. Value of buffer size and throughput.

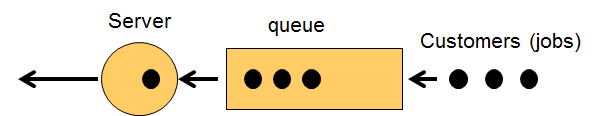
**2. Output:**

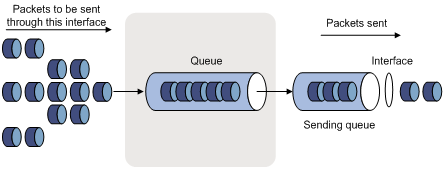
1. Communication time statistics of data packets.
2. Average of communication time or latency of this sample.

**Primary Metrics of Interest**:

1. **Latency**,
2. **Average buffer occupancy**,
3. **Average packet drop/ retransmission**.

**3. Content:**

1. **Entities:**
   1. **Consumer Entity**: Data packets,
   2. **Resource Entity**: The nodes of the network i.e. Routers, Gateways, Switches etc. 
   3. **Queue Entity**: Queue at each signal
2. **Activities**: Traveling across the network by each consumer entity obeying the first hop routing algorithm at each node.



**4. Simplifications:**

1. We simplify the SUI by taking very small segment of computer network comprising 2-3 nodes and 5-8 computers.
2. We assume the interface at each node be the same with same buffer size and same throughput.
3. Data packets are generated at each computer from i.i.d process.

5. **Assumptions:**

Typically in computer networks there are four types of Packet Delay

1. Nodal Processing
2. Queuing Delay
3. Transmission Delay
4. Propagation Delay

We assume for the simulation purpose

1. Delay is caused only due to queuing delay and fixed propagation delay between nodes.

**Simulation Steps**

1. Generate the data packets at each computer from exponential distribution of fixed rate.
2. Generate the destination of each data packets by random selection of IP address of any other computer on the network.
3. Attach data packets the IP address of origin and destination along with the time stamp.
4. At each node use the first hop routing algorithm to serve the data packets based on the IP address of origin and destination.
5. Dropping of packets at each node can be determined based on Active Queue Management Algorithms under investigation.
6. Retransmission of packets can be implemented using acknowledgement messages.
7. Fixed propagation delay between two nodes (Logical Processors) is used as Look Ahead in order to implement the Parallel and Distributed simulation using Chandy-Misra\_Bryant Algorithm.

**Description of simulation software**

The simulation software is developed in Python

**Working Simulator Code Submitted**

The bare bone working simulation code is submitted that simulates data communication on computer networks.

**Literature References**

1. L. G. Birta and G. Arbez, Modeling and Simulation: Exploring Dynamic System Behavior, Springer, 2007.
2. Bertsekas, Dimitri, and Robert Gallager. *Data Networks (2nd Edition)*. Upper Saddle River, NJ: Prentice Hall, 1991. ISBN: 0132009161.
3. Stevens. *TCP/IP Illustrated*. Reading, MA: Addison-Wesley Pub. Co., c1994-c1996. ISBN: 0201633469.
4. Kleinrock, Leonard. *Queueing Systems, Vol 1: Theory*. New York, NY: Wiley J., 1975. ISBN: 0471491101.