Text

Description automatically generated

The simulation has the following parameters

* avgPktLength: The average length of a packet
* rho: The rho parameter specified as the utilization of the queue
* serviceRate: The rate of which a packet is serviced in bits/sec
* queueSize: The size of the queue, which is either some number k or infinity
* maxTime: The time that the simulation should run for

The following variables are then also defined

* arrPktRate: The arrival rate of packets
* prevArrv: The time of the previous packet arrival
* prvDpt: The time of the previous packet departure
* packets: The list of packets with the information of time of arrival, packet length, service time, and time of departure
* currArrv: The time of the current packet arrival
* currDpt: The time of the current packet departure
* pktLength: The length of the packet
* serviceTime: The time it takes to service a packet

Text

Description automatically generated

This loop generates the arrival packets and its associated lengths and times to use as an event. The time between arrivals is a Poisson distribution, as well as the length of the packet. The time to service the packet is then calculated. The departure is calculated as the time of arrival plus the service time if the queue is empty, or the previous departure time plus the service time if the queue is not empty. The packet information is then added to the list and the previous arrival and departure times are updated for the next packet. The loop terminates once a packet arrival time passes the maximum simulation time.

Text

Description automatically generated

* observers: The list of observers
* prevObs: The time of last generated observer
* currObs: The time of the current observer

A list of observers is generated, with the time between observers being a Poisson distribution at 5 times the rate of the arrivals.

Text

Description automatically generated

* events: The list of events

Events is a list of arrivals, with its arrival time and service time, pre-generated departures, and observers. The events are sorted by the time of occurrence.

Graphical user interface, text

Description automatically generated with medium confidence

* c\_arr: The count of total arrivals so far
* c\_dpt: The count of total departures so far
* c\_obs: The count of total observers so far
* c\_idle: The count of the number of times the queue was observed to be empty
* c\_queue: The sum of all the lengths of the queues so far
* c\_loss: The count of total packets lost so far
* dpts: The list of departures and their departure time
* prevDpt: The time of previous departure

Text

Description automatically generated

The simulation runs differently depending on if the queue is infinite or not. If the queue is infinite, it will only look at the pre-generated departure times. If the event is an arrival, the number of arrivals increments by one. If the event is a departure, the number of departures increments by one. The observer notes that it is idle if the departures equals the arrivals, else it will increment the queue count by the difference.

For a finite queue, each event triggers a check on the list of departures. If any departure occurs before the event and has not been taken into account, then the departures are taken care of. This means that the pre-generated departure times are ignored. An arrival generates a departure event with a departure time depending on if the queue is empty, the arrival time, and the service time. If the queue is full, the packet is dropped. Observers behave the same way regardless of queue size.

Text

Description automatically generated

After the events terminate, if any departure occurs before the maximum simulation time is reached then they are also noted.

The values for average packets in queue, the loss rate, and idle rate are calculated and returned.

Text

Description automatically generated