

Laboratory 2

M/M/1 Queue simulation

Objective

This laboratory is important for understanding OPNET system and user interface. The lab contains a step-by-step example that shows how to use OPNET to construct an M/M/1 queue design and analysis.

Overview

The task is to construct an M/M/1 queue model and observe the performance of the queuing system as the packet arrival rates, packet sizes, and service capacities change. Two classes of statistics will be measured, Queue Delay and Queue Size. A graph of the confidence interval will also be produced.

This laboratory also introduces the use of:

- Node Model
- Probe Model
- Simulation Tool
- Analysis Tool

Procedure

An M/M/1 queue consists of a First-in First-Out (FIFO) buffer (queue) with packet arriving randomly according to a Poisson arrival process, and a processor, that retrieves packets from the buffer at a specified service rate. Three main parameters affects the performance of an M/M/1 queue, namely:

- Packet arrival rate, λ
- Packet size, $1/\mu$
- Service capacity, C

[7]

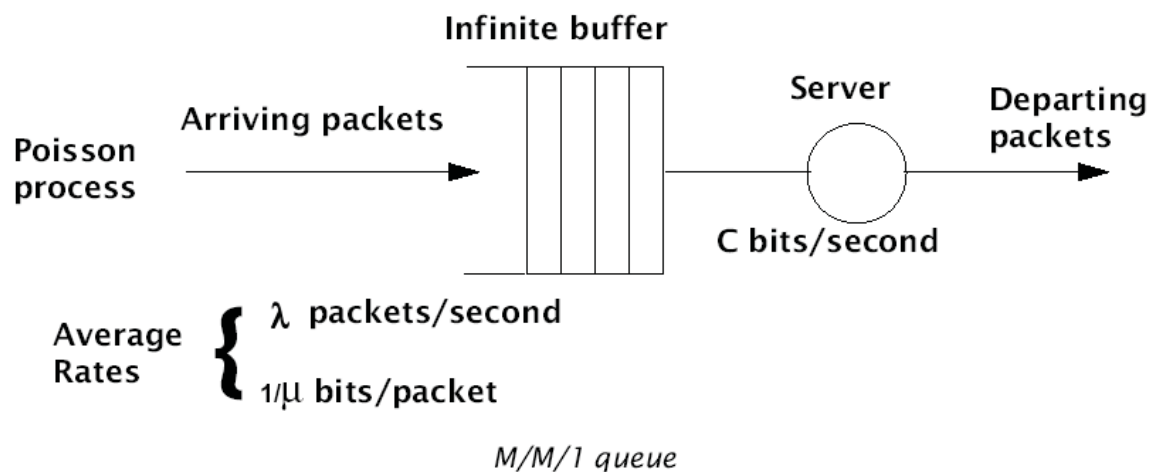


Figure 40 - M/M/1 overview

OPNET models are hierarchical. At the lowest level, the behavior of an algorithm or a protocol is encoded by a state/transition diagram, called state machine, with embedded code based on C-type language constructs. At the middle level, discrete functions such as buffering, processing, transmitting and receiving data packets are performed by separate objects. Some of these objects rely on underlying process models. In OPNET, these objects are called modules and they are created, modified and edited in the Node Editor. Modules are connected to form a higher-level node model. At the highest level, node objects are deployed and connected by links to form a network model. The network model defines the purpose of the simulation.

The lower-level objects for M/M/1 queue are provided by OPNET, but they need to be combined to form a node model.

Creation of the node model

The M/M/1 queue consists of several objects in the Node Editor, namely 2 processors and a queue. One processor is used for (source) packet generation. The second processor is used for the sink module. The source module generates packets and the sink module disposes the packets generated by the source. The queuing system is composed by an infinite buffer and a server.

The source module generates packets at an exponential rate.

- 1) Start OPNET.
- 2) Open a new project and a new scenario. Name the new project **<initials>_mm1net** and the scenario **MM1**.
- 3) Click **Quit** in the wizard.
- 4) Select **File => New...** , then select **Node Model** from the pull-down list and click **OK**.

The Node Editor appears.



- 5) Click on the **Create Processor** button.
- 6) Place a processor module in the workspace. Right-click to end the operation.
- 7) Right-click on the processor and select **Edit attributes**.
- 8) Change the **name** attribute to **Source**.
- 9) Change the **process model** attribute to **simple_source**.

Some new generator attributes appears in the attribute list.

- 10) Left-click in the Value Column of the **Packet Interarrival Time** attribute to open the **Packet Interarrival Time Specification** dialog box.
- 11) From the **Distribution Name** pop-up menu select **Exponential**, as in a Poisson process.
- 12) Set the **Mean Outcome** to **1.0** and click **OK**. This sets the mean interarrival time, λ , of a packet to 1 second.

- 13) Change the **Packet Size** attribute so that **Distribution Name** is **exponential** and **Mean Outcome** is **1024**. Click **OK**.

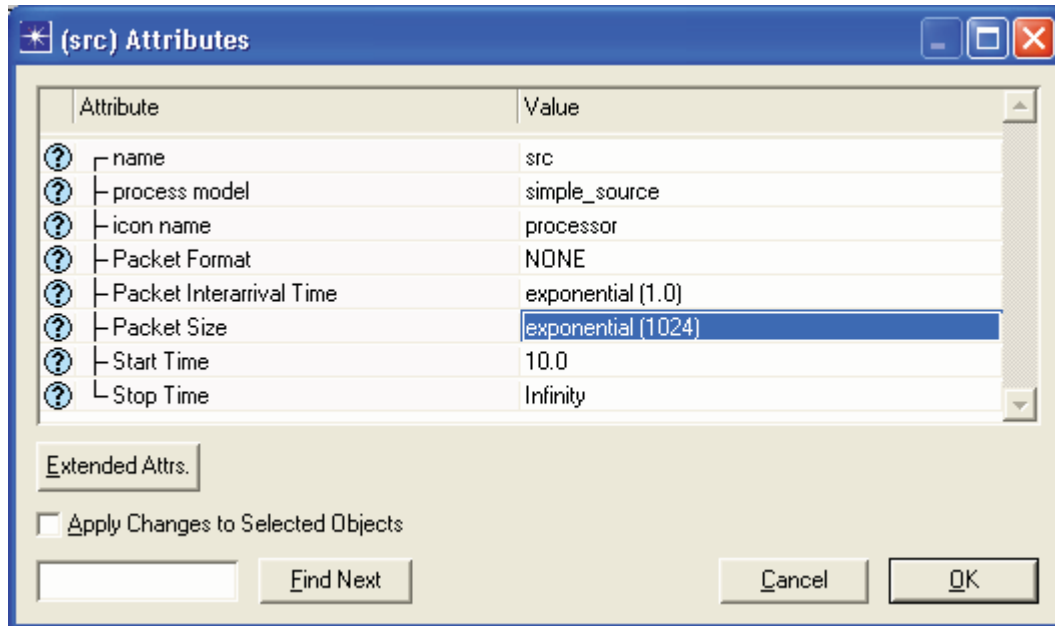



Figure 41 - Processor Model Attributes

- 14) Click **OK** to close the Attributes dialog box.
 In your design you will somehow have to represent the infinite buffer and the server, this will be done by the **queue** module. 

- 15) To create the queue model click first on the **Create Queue Module** button and place then a module in the workspace to the right of the generator.
- 16) When done right click to end the operation.
- 17) Further right-click on the queue module and select **Edit Attributes** to bring up its attributes.

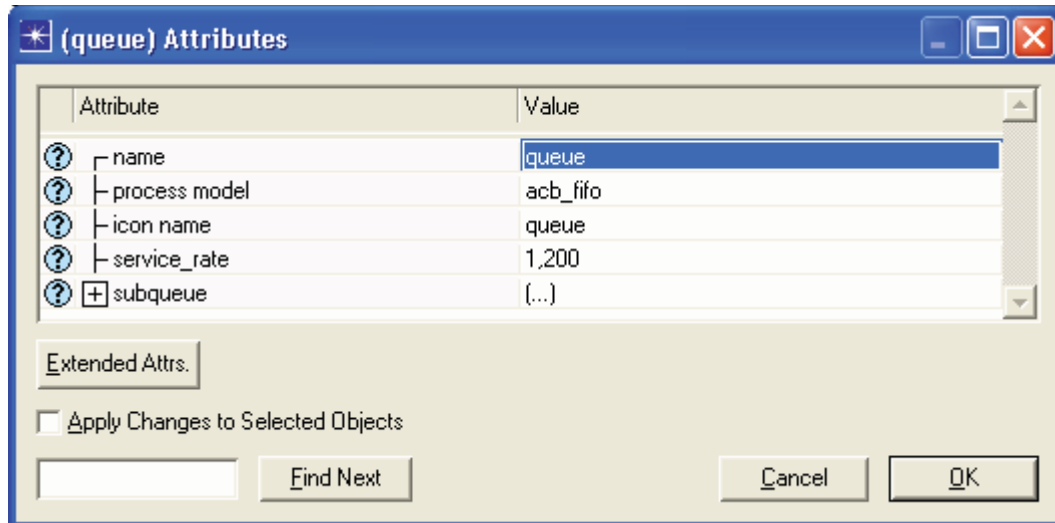


Figure 42 – Queue Process Model Attributes

- 18) Change the **name** attribute to **queue**.
- 19) Change the **process model** attribute to **acb_fifo**

The **acb_fifo** process is an OPNET-supplied process model that provides service in the packets arriving in the queue according to FIFO discipline. Note that when a process model is assigned to a module, the process model attributes appear in the module's attribute menu. The **acb_fifo** process model has an attribute called **service_rate**. When you select the **acb_fifo** process model, the **service_rate** attribute appears in the queue module's attribute menu with the default value 9600 bits per second. The name of the model **acb_fifo** reflects its major characteristics:

Characteristics	Indicates
A	Active, acts as its own server
C	Concentrates multiple incoming packet streams into its single internal queuing resource
B	Service time is a function of the number of bits in the packet
fifo	First in first out service ordering discipline

- 20) Right click on **service rate** in the attribute column.
- 21) Select **Promote attribute to higher level**. The value for this will be set later in the **Simulation Set** dialog box.
- 22) Click **OK** to close the dialog box.

Further a sink module should be used although it is not a part of the M/M/1 queue system. The reason for using a sink is for proper memory management, packets should be

destroyed when no longer needed. The OPNET-supplied sink process module destroys the packet sent to it.

- 23) To create the sink module, activate the **Create Processor Module** button and then place a processor module to the right of the queue model on the workspace.
- 24) Open the attribute box for the processor module by right clicking the icon.
- 25) Change the **name** attribute to **sink**. Notice that the default value for the **process model** attribute is **sink**.
- 26) Close the dialog box.

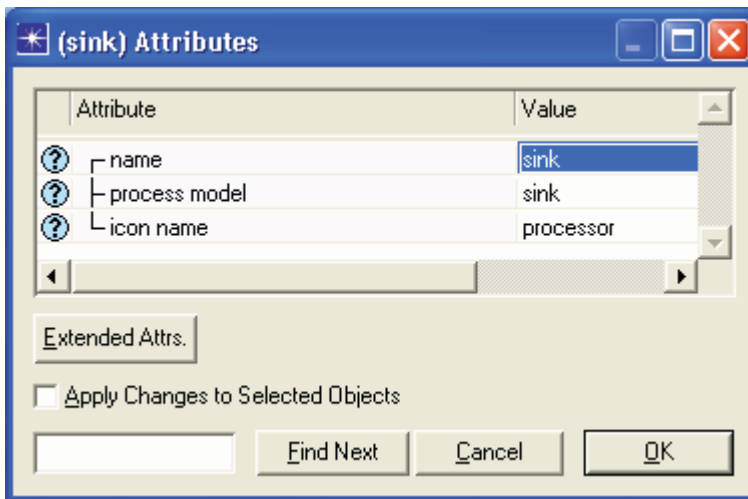


Figure 43 - Processor Model Attributes

To transfer packets between the generator module, the queue module and the sink module, it will be necessary to connect them together. This is done by packet streams that provide a path for the transfer of packets between modules. They serve as one-way paths and are depicted by solid arrowed lines.

- 27) Activate the **Create Packet Stream** button.
- 28) Connect the **source** module with the **queue module** by clicking the source icon, then clicking the queue icon. Remember that the first module you click becomes the source, and the second one the destination.
- 29) Connect the queue module and the sink module. Remember to end the **Create Packet Stream** operation by right-clicking anywhere in the window.
- 30) Select **Interface => Node Interfaces**.

Node Types		
Node Type	Supported	Default Icon
fixed	yes	fixed_comm
mobile	no	
satellite	no	

Figure 44 - Node Types dialog

31) In the **Node Types** table, click in the **Supported** cell for the **fixed** node type. This toggles the value to yes. Make sure mobile and satellite nodes are not supported.

32) Click **OK** to close the **Node Interfaces** dialog box.

The M/M/1 node model definition is complete. You are now ready to create the network model, but first you should save your work.

33) To save select **File => Save**. Name the node **<initials>_mm1**, then click **OK** and close the Node Editor.

The first step in creating the network model is to create a new model list. You can customize the palette to display only the models needed.

34) Click on **Open Object Palette** button.

35) Click on the **Configure Palette** button in the object palette.

36) Click on the **Model List** radio button.

37) Click on the **Clear** button.

38) Click on the **Node Models** button.

A list on available node models appears.

39) Scroll in the table until you find the **<initials>_mm1** node model. Change the **status** from **not included** to **included**.

40) Click **OK** to close the table.

41) Click the **Save** button. Enter the name **<initials>_mm1_palette**.

42) Click **OK**.

The node model you created is now included in the object palette.
Now you can create the network model.

- 43) Click and drag the **<initials>_mm1** node model to the workspace. Right-click to end the operation.
- 44) Change the name. Right-click and select **Set Name**.
- 45) Enter the name **m1** and choose **OK**.

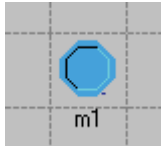


Figure 45 - M/M/1 Node Model

- 46) Save your model. **File => Save**.

We will use the Probe Editor to set the statistics to collect during simulation. This could also be done by the Choose Result option in the object pop-up menu. The Probe Editor is a more powerful collection tool.

The Probe Editor will be used to collect queue delay and queue size statistics.

- 47) Select **File => New...**, then select **Probe Model** and click **OK**.

You must specify the network model from which the Probe Model should collect statistics.

- 48) Select **Objects => Set Network Model**.
- 49) Choose the **<initials>_mm1net-mm1** network model.

We will now set the probes. Specifying appropriate probes causes the statistics to be recorded at simulation time.

- 50) Click the **Create Node Statistics Probe** button  and a probe appears below.

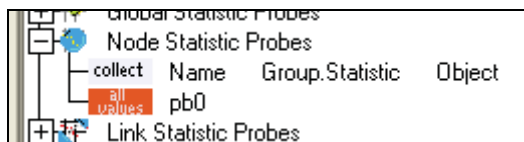


Figure 46 – Probe Attributes

We will use the automatic selection method to type in the attributes for the probe.

- 51) Right-click on the probe and select **Choose Probed Object** from the pop-up menu.
- 52) Expand the m1 node.
- 53) Select the **queue** module and click **OK**.

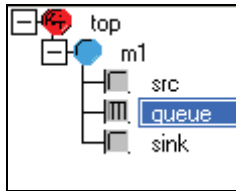


Figure 47 - Probe Objects

top.m1.queue appears in the Object column.

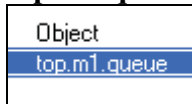


Figure 48 - Probe Object


- 54) Right click on the probe and select **Edit Attributes** from the pop-up menu.
- 55) Set the **name** to **Queue Delay**.
- 56) Set the **submodule** attribute to **Subqueue[0]**.
- 57) Left-click in the Value column of the **statistics** row.

The **Available Statistics** dialog box shows the statistics, the group it belongs to and a description.

- 58) Select **queue.queuing delay** from the list and click **OK**.

The group attributes changes to **queue** and the **statistics** attribute changes to **queuing delay** in the Edit Attributes dialog box.

- 59) Set **scalar data** to **enable**.
- 60) Set **scalar type** to **sample mean**.
- 61) Set **scalar start** to **14400**. This is done to eliminate the unwanted initial oscillations.
- 62) Set **scalar stop** to **18000**. This equals one hour.

- 63) Set **capture mode** to **all values**.
- 64) Click **OK** to close the probe's Attribute dialog box.
- 65) Click the **Create Node Statistics Probe** button  and a probe appears below.
- 66) You will now have to set new values to the attributes of the probe, to do so right click on the new probe and select **Edit Attributes**.

The values should be set as follows:

- **name = Queue Size**
- **subnet = top**
- **node = mm1**
- **module = queue**
- **submodule = subqueue [0]**
- **group = queue**
- **statistic = queue size (packets)**
- **scalar data = enabled**
- **scalar start = 14400**
- **scalar stop = 18000**
- **capture mode = all values**

- 67) When done close the probe's Attribute dialog box by clicking **OK**. The changes will then appear in the Probe Editor.

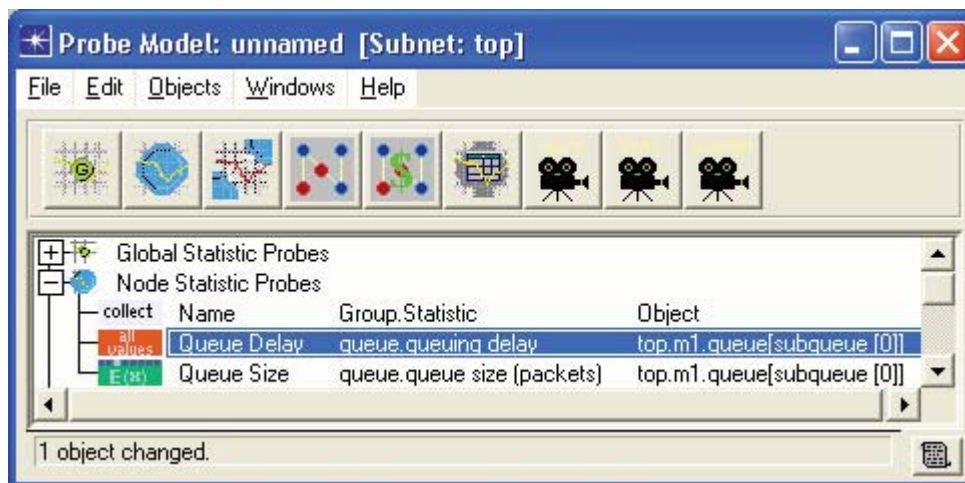



Figure 49 - Probe Model

- 68) Click on the **create attribute** button .
- 69) Right click and select **choose attributed object** in the pop-up menu.

- 70) Select **top** => **mm1** => **queue** then click **OK**.
- 71) Right click on the attribute probe and select **edit attributes**.
- 72) Set **attribute** to **service_rate**.

The probes are now set up correctly and will collect the desired statistics.

- 73) Save the probe file. **File** => **Save**. Name it **<initials>_mm1probe**.
- 74) Close the Probe Editor.

It's time to begin the simulation. We will use the Simulation Sequence Editor instead of running the simulation from the Project Editor.

- 75) To open the Simulation Sequence Editor choose **Simulation** => **Configure Discrete Event Simulation (Advanced)**.
- 76) Now place the simulation set icon in the workspace and change its values by selecting **Edit Attributes**.
- 77) The simulation **Duration** should be set to **7 hours** and the **Seed** to **321**.
- 78) Change the **value per statistic** to **10000**.

The seed is an initial value for the random generator. You can choose your own arbitrary value.

- 79) Now select the **Advanced** tab and then set the **probe file** to **<your initials>_mm1probe**.
- 80) Set **Scalar File** to **<initials>_mm1scalar**.
- 81) Select the **Object Attributes** tab.
- 82) Click the **Add** button.

An Add Attributes dialog box appears.

- 83) Mark **Add** in the column next to **m1.queue.service_rate**.
- 84) Click **OK** to close the dialog box.
- 85) Select the **m1.queue.service_rate** attribute in the Simulation Set dialog box and click the **values** button.

An Attribute: m1.queue.service_rate dialog box appears.

86) Set the **value** to **1050**.

87) Set the **Limit** to **1200**.

88) Set **Step** to **50**.



Figure 50 - Simulation Attributes

89) Click **OK** to close the Attribute: **m1.queue.service_rate** dialog box.

Notice that the number of runs has changed to 4.

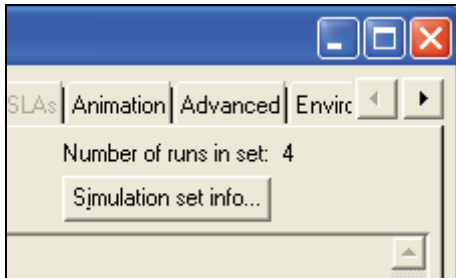


Figure 51 - Simulation dialog

90) Click **OK** to close the Simulation Set dialog box.

91) Click on the **scenario** to mark it.

92) Select **Edit => Copy** in the menu.

93) Select **Edit => Paste**.

94) Place the new scenario to the right of its parent.

95) Right click on the new **scenario** and select **Edit Attributes**.

96) Change the **Seed** number to a different value, when done click **OK** to close the dialog box.

97) Create eight more scenarios by repeating steps 91 to 96.

Notice: It's very important that each scenario has different Seed numbers.

98) When finished creating all the scenarios begin the simulation by clicking on the **Execute Simulation Sequence** button. Wait until the simulation is completed.

In this last portion of lab you will learn how to view your results. The tool used for this task is the Analysis Tool.

99) Select **File => New** from the Project Editor.

100) From the pull-down menu in the dialog box change the file type to **Analysis Configuration**, then click **OK**. The Analysis Tool opens in a new window.

During the simulation the packets always experience some delay. This delay is named the **mean queuing delay** and is the first statistic that we are going to take a closer look at.

101) To view the mean queuing delay left-click on the **create a graph of a statistic**



102) When done the View Results dialog box opens. Expand now the following hierarchy: **File Statistics => <your initials>_mm1net-mm1 => Object Statistics => mm1 => queue => subqueue [0] => queue**.

103) Select the **queuing delay** statistic.

104) From the list of filters select **average**.

105) Finally to show the graph click on the **Show** button.

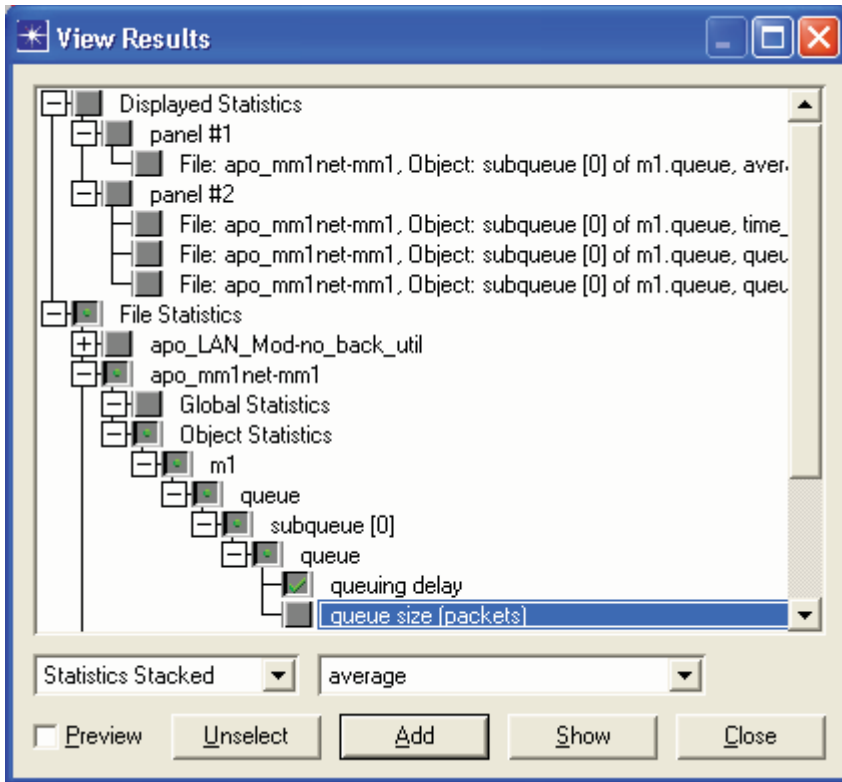


Figure 52 - View results dialog

Compare your graph with the one below, and see if it matches.

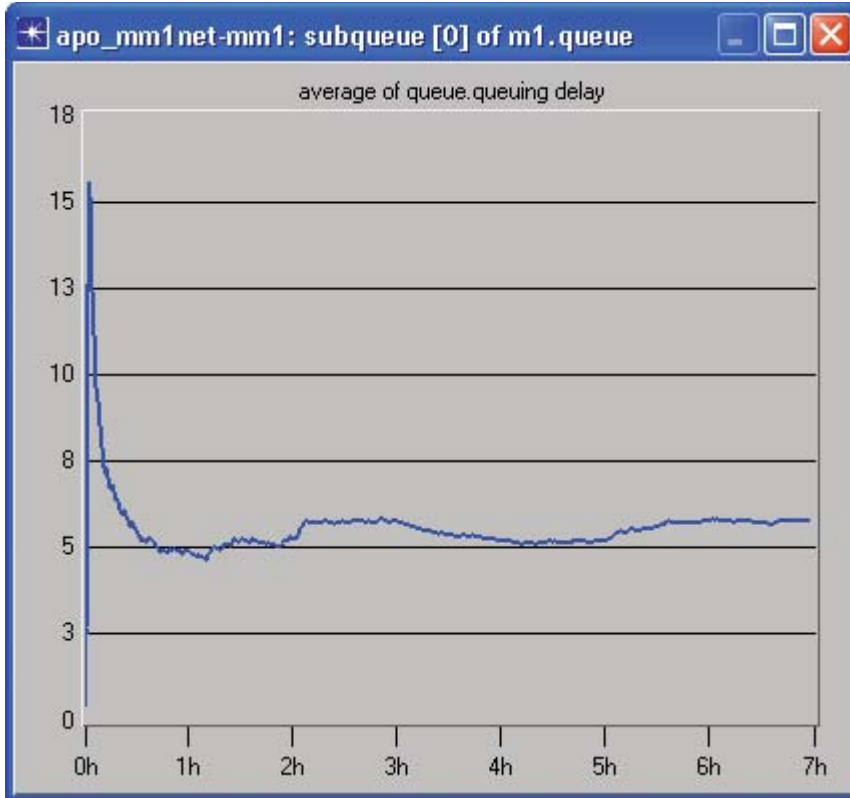


Figure 53 - Queue Delay graph

The large deviations early in the simulation depends on the sensitivity of averages to the relatively small number of samples collected, as you can see the average stabilizes towards the end of the simulation.

The results need to be validated.

Calculate the following:

Mean arrival rate, $\lambda = \frac{1}{\text{mean interarrival time}} =$

Mean service requirement, $\frac{1}{\mu} =$

Service capacity, $C =$

Mean service rate, $\mu C =$

Mean Delay, $\bar{W} = \frac{1}{\mu C - \lambda} =$

Does the analytic value agree with the simulated result?

Another statistic of interest is the time-averaged queue size.

106) In the View results dialog box, remove the check next to the **queuing delay**.

107) Place a check next to the **queue size (packets)** statistic.

108) Select **time-average** from the pull-down list.

109) Click **Show**.

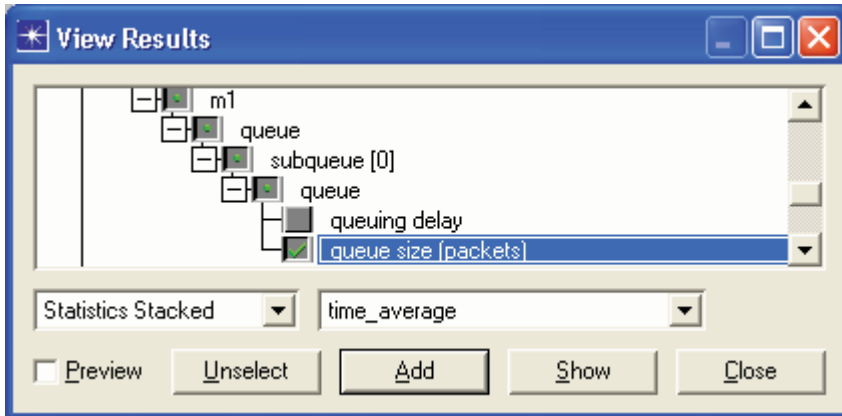


Figure 54 - View Results dialog

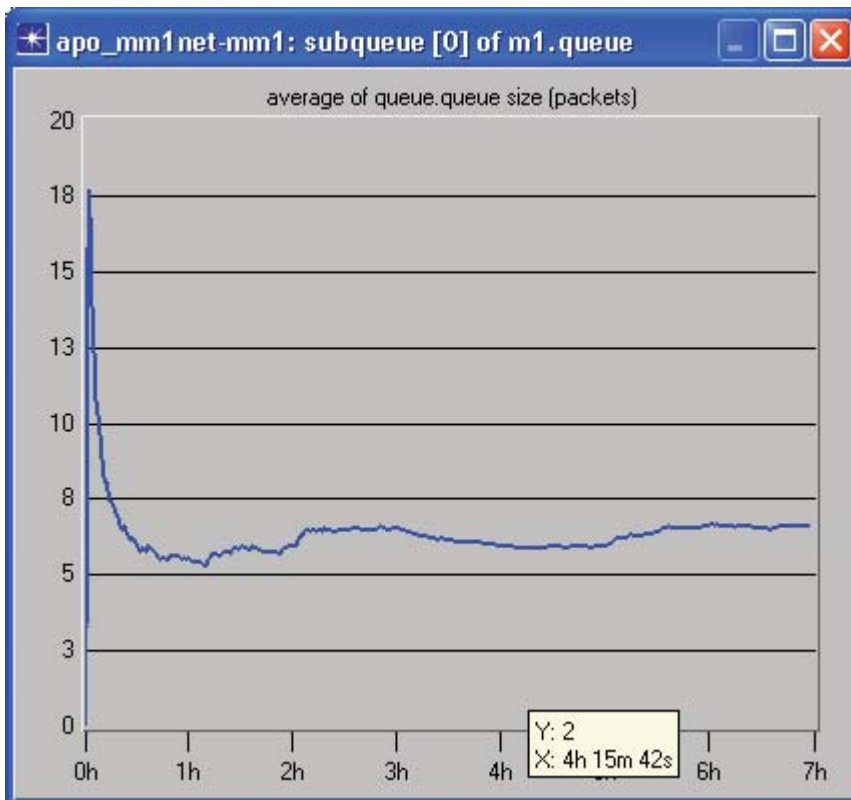


Figure 55 - Queue Size graph

From this graph we can draw the conclusion that the system is stable. It reaches steady state after about 2 hours.

Validate the result.

Calculate the following:

Ratio of Load to Capacity $\rho = \frac{\lambda}{\mu C} =$

Average Queue Size $\frac{\rho}{1 - \rho} =$

Does the analytic value agree with the simulated results?

We will produce a queue size versus time averaged queue size graph.

- 110) Set focus on the **time averaged queue size** window.
- 111) Right-click on the graph and choose **Add statistics**.
- 112) Place a check next to **File Statistics => <initials>_mm1net-mm1 => Object Statistics => m1 => queue => subqueue [0] => queue => queue size (packets)**.
- 113) Left-click on the **Add** box.
- 114) Save the analysis configuration as **<initials>_mm1net-mm1**.

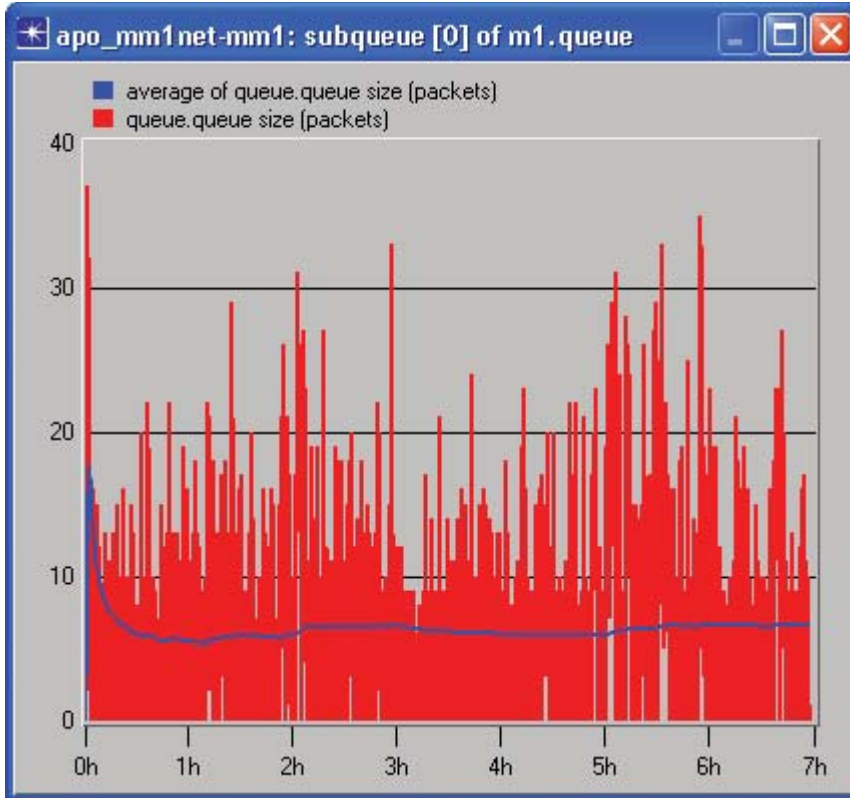


Figure 56 - Queue Size, "As Is" and "Average" overlapped graph

It's important to know the confidence intervals of the simulation results. We will now graph the confidence interval.

115) Load the Scalar file. **File => Load output scalar file...**

116) Select `<initials>_mm1scalar`.

117) Click the **Create graph of two scalars** button.



118) Select `top.queue.service_rate` on the **Horizontal** roll-down menu.

119) Select `top.queue.subqueue[0].queue.queue size (packets).mean` on the **Vertical** roll-down menu.

120) Right click on the graph and select **Edit Graph Properties**.

121) Check the **Show Confidence Interval** check box.

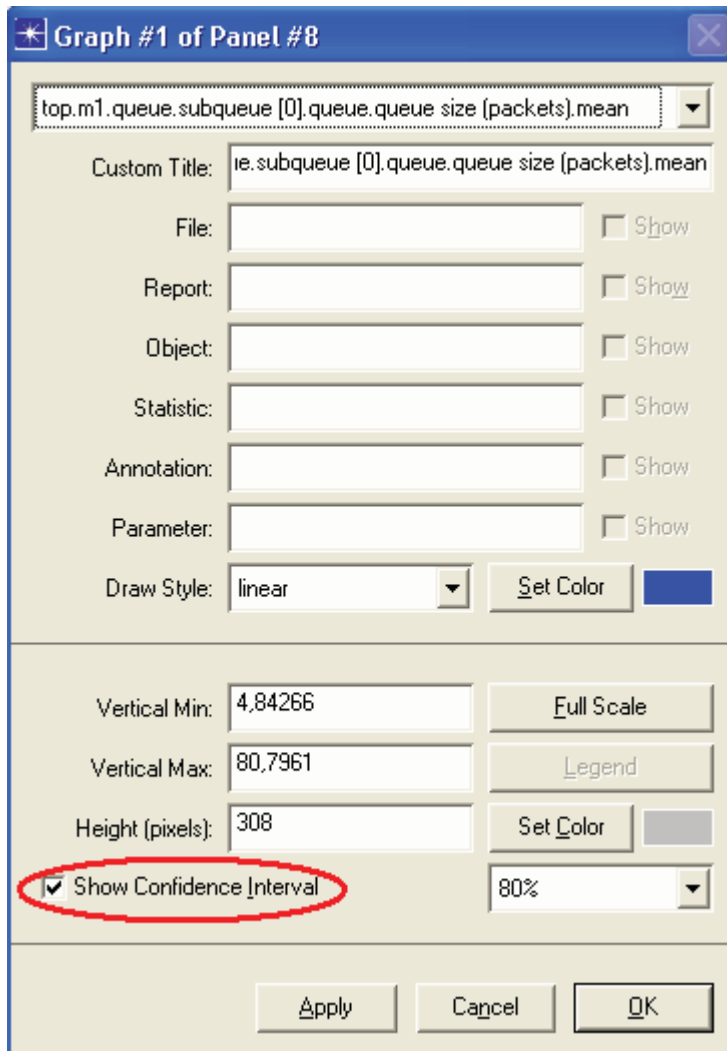


Figure 57 - Graph Plot properties

122) Click **OK** to close the dialog.

The graph should resemble the one below.

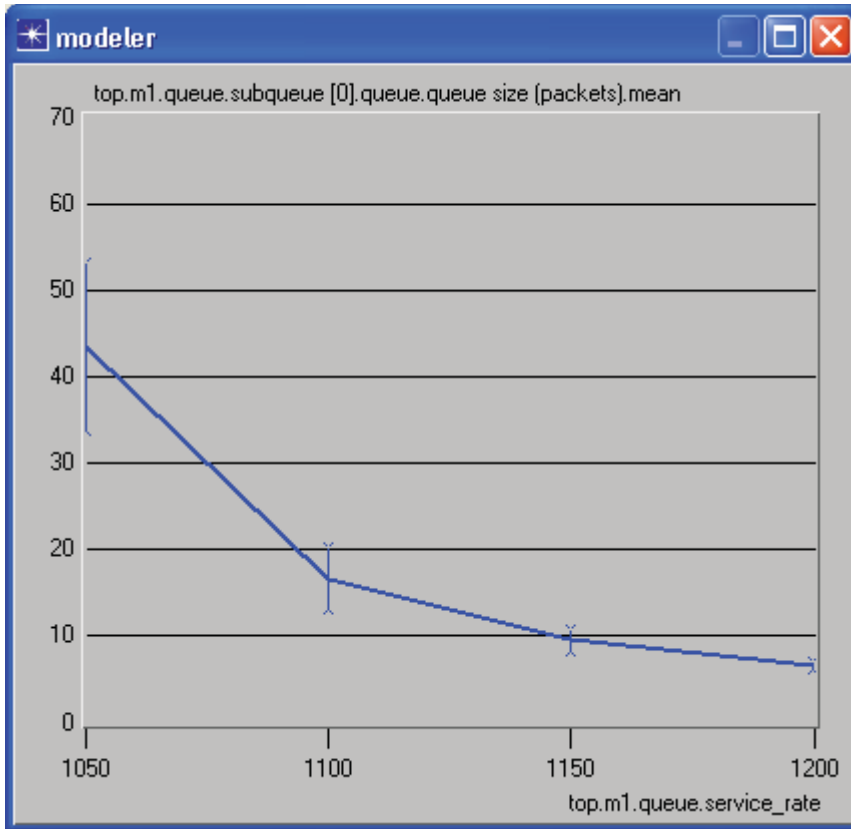


Figure 58 - Queue Size Confidence Intervall graph

In order to get better confidence intervals you need to run more scenarios. The graph below shows the confidence interval of 40 scenarios. The service rate is between 1024 and 1300 with step size 10.

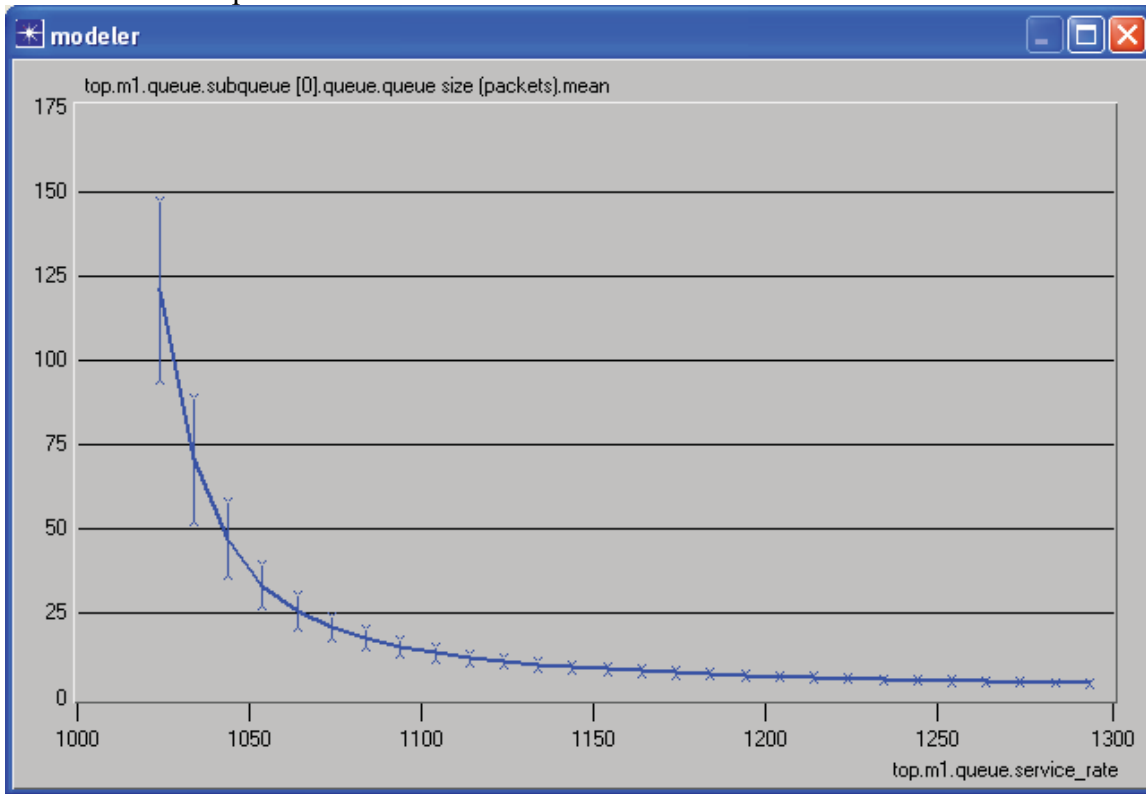


Figure 59 - Queue Size Confidence Interval graph

Congratulations!
You have finished the lab.