

# EE984 Laboratory Experiment 1: Network Simulator

## Table of Content

1. Introduction.....	2
1.1. About OPNET .....	2
1.2. Learning Outcomes.....	2
1.3. Assessment .....	3
2. Laboratory Session 1.....	3
2.1. How to Use OPNET .....	3
2.2. Part 1. Familiarisation with the OPNET simulator.....	3
2.3. Part 2. Examples of the use of OPNET .....	4
3. Laboratory Session 2.....	5
3.1. Introduction .....	5
3.2. Part 3 Exponential Sources.....	6
3.2.1. Background on Exponential Sources.....	6
3.2.2. Exponential Sources and OPNET .....	7
3.2.3. Analysis of Inter-arrival times .....	7

## 1. Introduction

### 1.1. About OPNET

A key feature of network design is the estimation of the performance of a proposed network design. In the case of realistic network topologies and network applications performance estimation is a difficult problem. As a result, a number of powerful simulators have been developed that are capable of attacking these problems. Two of the well-known simulators are called **Ns2** and **OPNET**.

Ns2 is a free simulator that you can download from the Web and use yourself (enter 'ns2' in the Google search engine ([www.google.com](http://www.google.com)) to find all the necessary information to do this). Berkeley University and others have developed it over a period of a decade or so. Ns2 is 'open source' software so you can look at the implementation and even contribute to the development of Ns2 yourself! Ns2 has been used to address some of the fundamental problems in networking (such as the 'self-similarity' of IP traffic, thought to be the result of the way that TCP controls traffic flow).

OPNET is a commercial simulator. So whilst you don't have access to the simulator kernel source code itself OPNET has a productive and user-friendly front end that allows the efficient simulation of large networks (as this laboratory seeks to demonstrate).

The laboratory has two parts. In the first part you will follow the tutorials created by the vendors of OPNET in order to master the basic operation of the simulator. In the second part you will validate the results from the simulator using the mathematics of the M/M/1 queue. You will perhaps have not come across the M/M/1 queue but do not worry! A prime aim of this laboratory is to familiarise yourself with some of the basic concepts of queueing theory so that you will better appreciate them when they are introduced in the EE904 lectures later this term.

**OPNET Floating Licences:** The OPNET software is protected by a system of floating licences. Given the limited number of these licences in the department it may be necessary for you to work in groups during the formal laboratory session. However, it should be possible to work individually outside of these hours (assuming not too many people want to do so at the same time – this itself is a queueing problem!).

### 1.2. Learning Outcomes

**LA1** Learn how to use the OPNET network simulator to simulate a network.

**LA2** Learn how to validate the OPNET simulator by comparing simulator output with known theory.

**LA3** Learn how to check the mathematical properties of a fundamental M/M/1 queueing system using OPNET.

### 1.3. Assessment

You are to submit written accounts of the tasks as described in the script. Take care to follow the guidelines for report writing given before at the beginning of this laboratory experiment.

You are to write up tasks 1-6 in the way indicated. Marks will be awarded as follows:

**60% Technical:** Does the report describe a successful completion of the experiment and indicate that the author has understood how to use OPNET and also mastered the mathematical ideas behind the simulations? Has the statistical analysis been carried out correctly and the conclusions correctly drawn? Has the author indicated some creativity in his/her carrying out of the experiments?

**40% Style:** Are the experiments written up in the expected manner? Is the use of English language adequate and the spelling correct? Is the English concise and to the point? Is the material logically structured? Are diagrams used to good effect. Is the report below the maximum page count? Are references correctly used?

## 2. Laboratory Session 1

### 2.1. How to Use OPNET

This session has three parts, which should be completed before the next week laboratory session.

**To Start the OPNET Simulator:** The OPNET simulator is available in the software laboratory 4SB.6.8B on level 6 in the Department of Electronic Systems Engineering. The simulator is on all machines and runs under Windows, so when you turn it on select **win2k** as the operating system. Select **START|PROGRAMS|OPNET8.1|startup.bat**. If all goes well OPNET will start and you will get the **OPNET start-up window**. If it does not, please **ask a demonstrator** for help.

**To Finish a Session on the OPNET Simulation:** Your work (called a **project** in OPNET) is saved in a directory called **C:\documents and settings\username\local\temp\op\_models** where 'username' is your username. You should make a copy of this directory on your M: drive. When you return to the project later recopy the directory into the above directory. **You must do this as the C: drives are regularly 'cleaned up'** so you will lose your work if it is not saved in this way. This process may be automated for you.

### 2.2. Part 1. Familiarisation with the OPNET simulator

OPNET comes with an on-line tutorial. On the start-up window select the '**Online Tutorial**' item in the '**Help**' menu. The tutorial is a 'pdf' document so, if you wish, you can take a copy of it and print off relevant sections.

Select the '**Online Tutorial**' item to display the list of lessons in the tutorial. In this laboratory you are required to start by reading the '**Introduction**'.

Your first task is to read the Introduction. **In fact it might be a good idea to do this before going to the laboratory.** This will give you an overall picture of the OPNET architecture.

**Task 1.** Using the ‘Introduction’ as the source material, produce a one-page executive summary of the OPNET architecture and its capabilities. This summary must be **no longer than one A4 page**. The intended recipient is to be senior management. You may assume they are knowledgeable network engineers but that they have no knowledge of OPNET.

### **2.3. Part 2. Examples of the use of OPNET**

Now that you have a broad idea of the capabilities of OPNET you can learn how it is used in detail to solve a realistic network problem. For this, select the ‘**Small Internetworks**’ lesson from the on-line tutorial. Work through this tutorial. This may take you an hour or more and it might be a good idea to go through the tutorial a second time to reinforce your understanding of the use of OPNET.

**Task 2.** Describe the problem to be solved in this exercise and the steps required in the OPNET simulator to solve it. Your description is to be no more than **1 A4 page in length**.

Now work through the tutorial ‘**M/M/1 Queue**’.

Works through ‘**M/M/1 Queue**’ again but this time omit the sink,

**Task 3.** Write up the both simulations of the M/M/1 queue (**2 A4 pages maximum**) as a mini-experiment. Take care to try to explain what happened in the second simulation without the sink.

**Tasks 1, 2 and 3 should be completed in week 2 (using time outside the laboratory if necessary).**

**[End of Laboratory Session 1]**

### **Further Study of OPNET**

As you probably already appreciate, OPNET is a large and sophisticated simulator that is capable of simulating many realistic scenarios to be found in today’s networks. You may choose to investigate the simulator further for your own benefit. Perhaps you might be able to use it in your project work or to help you understand other aspects of the course.

## 3. Laboratory Session 2

### 3.1. Introduction

In last week's laboratory you learned the mechanics of the use of the OPNET simulator. In this week's session you will validate the simulator itself.

There is a considerable lore in the network (and other) simulation communities that go something like this. A speaker at a conference announces a switch design with spectacular performance. The switch performance was determined using simulation (using a proprietary commercial package so that it was not possible to look at the source to check that it was coded up correctly). The conference audience is incredulous, and rightly so, because the speaker later checked that the simulator was giving correct results – it wasn't. The speaker was understandably reluctant to attend the following year's conference! Message:- **Do your validation before you report simulation results – not after!**

**Validation** of a simulator is the process of checking that it is giving accurate results. Now it is not possible to independently validate all results of a simulator (if it were, there would be no need for the simulator in the first place!) but it is possible to check that it gives results that are known to be right from other sources. These other sources might include another independent simulator, in which case the agreement between simulators gives us confidence that both simulators are working correctly on the problem addressed. Thus Ns2 and OPNET could be used on the same problem to give us confidence in the simulated results.

Another source of known results is mathematical analysis. In rare cases mathematical analysis might be able to give the exact result to our problem (this is the case with the M/M/1 queue). In these cases we would expect the simulator to be able to reproduce these results given enough simulation time. If it cannot we would be unwise to continue to use the simulator.

In realistic scenarios however, mathematical analysis is only likely to give us **asymptotically** exact statements about the exact result (typically of the form: the result should fall off exponentially for large values of a system parameter, e.g., under certain conditions it is known that the packet loss probability in a router falls off exponentially in the buffer size for **large** buffer sizes). In this case the simulator should be able to give results that are close to the asymptotic regime for simulations that occupy this regime.

In short, because of the large number of possible ways complex simulations can go wrong it is vitally important that as many checks as possible be made on the result. **Forget this at your peril!** You may decide to use OPNET as part of your project work at Essex, in which case there is a good chance that you will be asked what steps you took to make sure the simulation results were accurate.

The module EE904-Network Engineering will provide you will some of the mathematical ideas required to be able validate network simulators in the way

described above. This second half of the experiment, in addition to validating OPNET, is intended to give you some practical experience with these mathematical ideas. Hopefully this will lead to an increased understanding of the ideas when they are formally presented to you.

## 3.2. Part 3 Exponential Sources

### 3.2.1. Background on Exponential Sources

In last week's lab you configured and used exponential sources. These model a packet source (e.g. a video or audio stream) as a sequence of packets with random temporal separation (or **inter-arrival time**) of the special form: the time  $T$  between two packets is a random variable with a probability distribution of the form  $P(T > a) = \exp(-\lambda a)$  (the left-hand side of this equation is the probability that the inter-arrival time between two packets is greater than a given value 'a').

**Question 1.** Explain why the units of  $\lambda$  are  $s^{-1}$ ? This is why  $\lambda$  is referred to as being the **source rate**.

**Question 2.** If  $\lambda = 1s^{-1}$  what is the probability that the next two packets from the source are separated by more than 5 seconds?

**Question 3.** Why is it true that  $P(T \leq a) = 1 - \exp(-\lambda a)$ ?

In the absence of other information the inter-arrival time between packets is often taken to be of this exponential form (in which case the total number of packets to arrival up-to time  $t$ ,  $N(t)$ , is called a sample path from a Poisson Process). In absence of other information, the size of each packet is often also taken to be exponentially distributed with a rate  $\mu$  or a mean temporal length of  $1/\mu$ . If we assume a router takes a time equal to the packet size (expressed in terms of its temporal extent) to 'serve' (i.e., send the packet off on the correct output link) that packet then we have an  $M(\lambda)/M(\mu)/1$  queue. The final '1' indicates that there is a single processor in the router doing the serving. If a packet is being served when another arrives then the latter has to wait in a queue for its turn to be served. An  $M/M/1$  queue is often represented diagrammatically as in Figure 1.

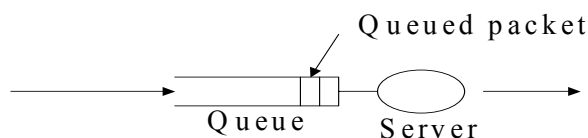


Figure 1.  $M/M/1$  Queue

### 3.2.2. Exponential Sources and OPNET

A simulator such as OPNET offering ‘exponential sources’ will have to be able to generate a sequence of inter-arrival times that come from the above distribution. However, we just want to verify that OPNET is correctly carrying out this task. How can we do this? There are a number of sophisticated methods to do this but this lab is not the place to study these methods. We will carry out some simple tests instead.

The first thing we need to do is to collect the inter-arrival times from OPNET. To do this carry out the following sequence of steps:-

- 1) Using the same procedure as last week create a node consisting simply of an exponential source and a sink into which it feeds.
- 2) Create a probe that measures the times at which the source sends a packet:
  - a. Probe the sink
  - b. Choose as statistic for the sink ‘packets received’
  - c. Select ‘Collect All Values’ on the menu that also contains the ‘Edit Attributes’ for the node probe
- 3) Run the simulation (choose a suitable time and a random seed so that you get enough packet transmission times to do analysis but not too many to make subsequent spreadsheet manipulations awkward)
- 4) Using ‘Analysis Configuration’ plot the ‘packets received’ statistic ‘As Is’.
- 5) Right click the plot to reveal a pop up menu and select ‘Export Graph To Spreadsheet’

If everything has worked you will now have an Excel spreadsheet giving you the time at which each packet was sent. Use a spreadsheet formula to create a column of inter-arrival times.

### 3.2.3. Analysis of Inter-arrival times

What simple tests can be carried out to see if the sequence of inter-arrival times has indeed come from an exponential distribution?

---

#### Aside

Strictly speaking, it would take an infinite number of tests to verify that the source is exponential. Given the way simulators usually generate probability distributions, with the use of a deterministic algorithm (such as a pseudo-random binary sequence (PRBS)), it is almost certainly true that the source is not exponential. However, from the simulation point of view it is sufficient that the source exhibit, within given tolerances, the required properties only, e.g., the correct inter-arrival time distribution needs to be ‘right’. There may be other moments of the distribution that are not correct but the people carrying out the simulation can perhaps live with that!

---

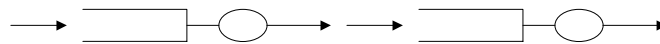
**Task 4.** Using your answers to questions 1,2 and 3 above, design and implement an experiment to test that the OPNET exponential source is indeed exponential. Carry out your statistical calculation in the Excel spreadsheet containing the raw data. Briefly write up this experiment (aim, method, results, conclusions). **Including relevant graphs the write-up should be no longer than 2 A4 pages.**

Ideally, we would like to know if our chosen simulator is efficiently implemented. Carry out the following task to try to find out.

**Task 5.** Implement a program (in a language of your choice) that generates exponentially distributed random numbers. Use your program and OPNET to generate the same number of random numbers. Which runs faster? (Hint: look carefully at the output given by OPNET for useful information). Write up task 5 as a mini-experiment. **Maximum page count is to be one A4 page.**

**Task 6. Simulate a Tandem Queue.**

As you are probably aware, communication networks can be thought of as a system consisting of many queues (the packet buffers in routers and switches) which are served by routers/switches that take packets/cells and pass them on over links to other queues (or perhaps to the final destination). Thus in a realistic network a packet is likely to have to pass through several queues before reaching its final destination – the simulation of a single M/M/1 queue is unlikely to give us a great deal of information about network behaviour. In this simulation you are to create a system of two identical servers in series (see Figure. 2). Use the same server queue as used earlier in the laboratory to determine a graphical display of the queuing delay as a function of time for each of the queues. You are also to investigate the inter-arrival time for packets arriving at the second queue. In particular you are to test the hypothesis that the distribution is the same as that of the source. Write up task 6 in the form of an experiment. **Maximum page count is to be four A4 pages.**



**Figure 2. Tandem M/M/1 Queues**

**[End of Laboratory 1]**