

**LAB MANUAL OF
COMPUTER NETWORKS LAB
ETCS 354**



Maharaja Agrasen Institute of Technology, PSP area,
Sector – 22, Rohini, New Delhi – 110085
(Affiliated to Guru Gobind Singh Indraprastha University,
New Delhi)

INDEX OF THE CONTENTS

- 1. Introduction to the lab**
- 2. Lab Requirements (details of H/W & S/W to be used)**
- 3. List of Experiments as per GGSIPU**
- 4. List of experiments beyond the syllabus**
- 5. Format of the lab record to be prepared by the students.**
- 6. Marking scheme for the Practical Exam**
- 7. Instructions for each Lab Experiment**
- 8. Sample Viva – Questions**

1. Introduction to the Lab

Lab Objective

The objective of this lab is making student know how to design and analyze computer networks. To become acquainted with network programming and some of the important GUI based computer networking tools.

Course Outcomes

At the end of the course, a student will be able to:

C354.1 Simulate the Discrete Event Systems using various network tools.

C354.2 Design solutions for real life situations in form of communication networks.

C354.3 Evaluate all the possibilities of wired as well as wireless networks (Zigbee, Wi-Max, Wi-PAN, IEEE 802.11 a,b,c,g) by using routers, switches and various topologies.

C354.4 Analyze and evaluate the network results using different open source logger tools (Wireshark, TCPDump and NS3 NetAnim package).

C354.5 Implement Sliding window and congestion avoidance protocols.

C354.6 Explore the possible research opportunities and difficulties within the course scope.

2. LAB REQUIREMENTS

Hardware Detail

Intel i3/C2D Processor/2 GB RAM/500GB HDD/MB/Lan Card/ Key Board/ Mouse/CD Drive/15” Color Monitor/ UPS	24 Nos
LaserJet Printer	1 No

Software Detail

Linux, Network Simulator v 2 & 3 and Wireshark.

3. LIST OF EXPERIMENTS

(As prescribed by G.G.S.I.P.U)

1. Introduction to Computer Network laboratory, introduction to Discrete Event Simulation and Discrete Event Simulation Tools
2. Introduction to NS3 and its comparison with NS2.
3. Install NS3 on Linux.
4. Using Free Open Source Software tools *ns3*, design and implement two nodes topology.
5. Using Free Open Source Software tools *ns3*, design and implement three nodes topology considering one node as a central node.
6. Using Free Open Source Software tools *ns3*, design and Implement star topology using StarHelperClass.
7. Using Free Open Source Software tools *ns3*, design and implement a bus topology using CSMA.
8. Using Free Open Source Software tools *ns3*, design and implement hybrid topology connecting multiple routers and nodes.
9. Install and configure NetAnim.

4. LIST OF EXPERIMENTS (Beyond the syllabus)

1. Using Free Open Source Software tools *ns3*, design and implement FTP using TCP bulk transfer.
2. Analyze network traces using Wireshark software

5. **FORMAT OF THE LAB RECORD TO BE PREPARED BY THE STUDENTS**

The front page of the lab record prepared by the students should have a cover page as displayed below.

NAME OF THE LAB

Paper Code

Font should be (Size 20", italics bold, Times New Roman)

Faculty name

Student name

Roll No.:

Semester:

Font should be (12", Times Roman)



Maharaja Agrasen Institute of Technology, PSP Area,

Sector – 22, Rohini, New Delhi – 110085

Font should be (18", Times Roman)

Index

Exp. no	Experiment Name	Date of performance	Date of checking	Marks	Signature

6. MARKING SCHEME FOR THE PRACTICAL EXAMS

There will be two practical exams in each semester.

- i. Internal Practical Exam
- ii. External Practical Exam

INTERNAL PRACTICAL EXAM

It is taken by the respective faculty of the batch.

MARKING SCHEME FOR THIS EXAM IS:

Total Marks: 40

Division of 10 marks per practical is as follows:

Rubrics for : Laboratory (General)				
Sr No.	Experiment Component (LAC)	Max. Marks	Grading Rubrics	
			2 marks	1 mark
1	Practical Performance	2	Completeness of practical, exhibits proficiency in using different types of inputs.	Incomplete practical, unformatted, lacks comments, Demonstrates no proficiency.
2	Output and Validation	2	Output is free of errors and output is obtained. Demonstrates excellent understanding of the concepts relevant to the experiment.	Output contains few logical errors and/or no output is obtained. Demonstrates partial understanding of the concepts relevant to the experiment.
3	Attendance and Viva Questions Answered	4	1. Four marks for answering more than 75% questions. 2. Three marks for answering more than 60% questions. 3. Two mark for answering more than 50% questions. 4. One mark for answering less than 50% questions.	
4	Timely Submission of Lab Record	2	On time submission	Late submission

Each experiment will be evaluated out of 10 marks. At the end of the semester average of 8 best performed practical will be considered as marks out of 40.

EXTERNAL PRACTICAL EXAM

It is taken by the concerned lecturer of the batch and by an external examiner. In this exam student needs to perform the experiment allotted at the time of the examination, a sheet will be given to the student in which some details asked by the examiner needs to be written and at the last viva will be taken by the external examiner.

MARKING SCHEME FOR THIS EXAM IS:

Total Marks: 60

Division of 60 marks is as follows

1. Sheet filled by the student:	20
2. Viva Voice:	15
3. Experiment performance:	15
4. File submitted:	10

NOTE:

- Internal marks + External marks = Total marks given to the students
(40 marks) (60 marks) (100 marks)
- Experiments given to perform can be from any section of the lab.

Introduction to Computer Networks Lab

The lab of Computer Networks gives in depth view of the computer networks working in real time and simulation of various topologies using NS3 tool.

ns-3 has been developed to provide an open, extensible network simulation platform, for networking research and education. In brief, *ns-3* provides models of how packet data networks work and perform & provides a simulation engine for users to conduct simulation experiments. Some of the reasons to use *ns-3* include to perform studies that are more difficult or not possible to perform with real systems, to study system behavior in a highly controlled, reproducible environment, and to learn about how networks work. Users will note that the available model set in *ns-3* focuses on modeling how Internet protocols and networks work, but *ns-3* is not limited to Internet systems; several users are using *ns-3* to model non-Internet-based systems.

Many simulation tools exist for network simulation studies. Below are a few distinguishing features of *ns-3* in contrast to other tools.

- *ns-3* is designed as a set of libraries that can be combined together and also with other external software libraries. While some simulation platforms provide users with a single, integrated graphical user interface environment in which all tasks are carried out, *ns-3* is more modular in this regard. Several external animators and data analysis and visualization tools can be used with *ns-3*. However, users should expect to work at the command line and with C++ and/or Python software development tools.
- *ns-3* is primarily used on Linux systems, although support exists for FreeBSD, Cygwin (for Windows), and native Windows Visual Studio support is in the process of being developed.

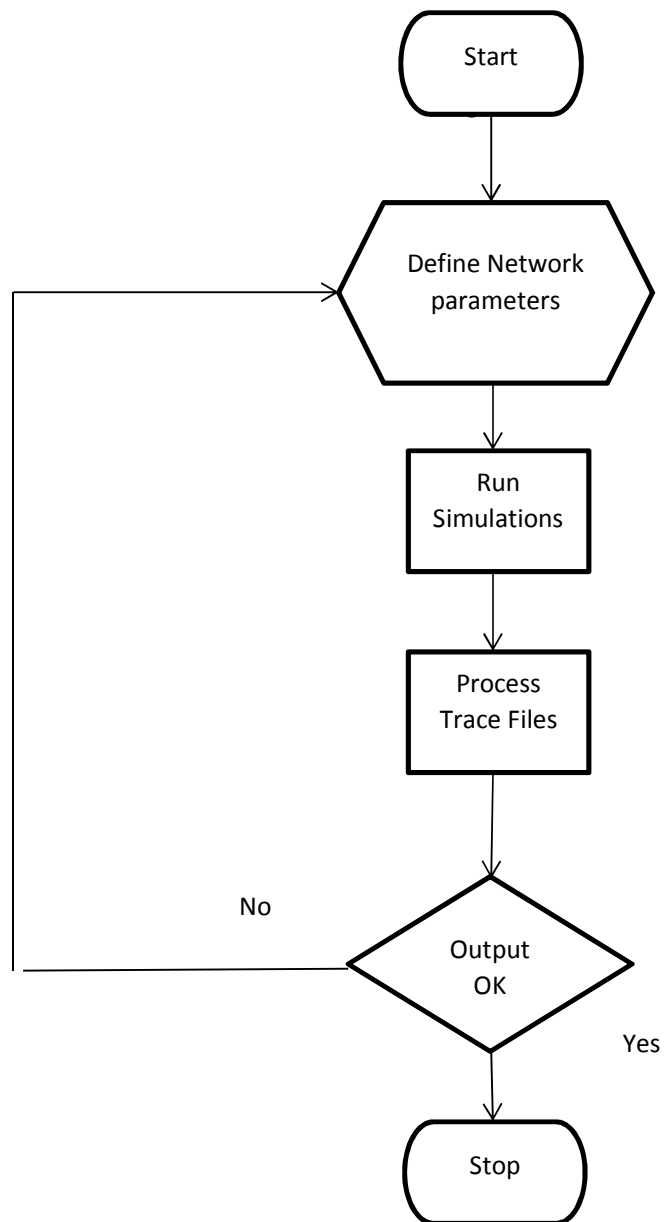


Fig 1: Simulation Process

INSTRUCTIONS FOR EACH LAB EXPERIMENT

EXPERIMENT 1

Aim: Introduction to Computer Network laboratory, introduction to Discrete Event Simulation and Discrete Event Simulation Tools

System:

A collection of entities that act and interact together toward the accomplishment of some logical end.

Discrete system:

State variables change instantaneously at separated point in time, e.g., a bank, since state variables - number of customers, change only when a customer arrives or when a customer finishes being served and departs

Continuous system:

State variable change continuously with respect to time, e.g., airplane moving through the air, since state variables - position and velocity change continuously with respect to time

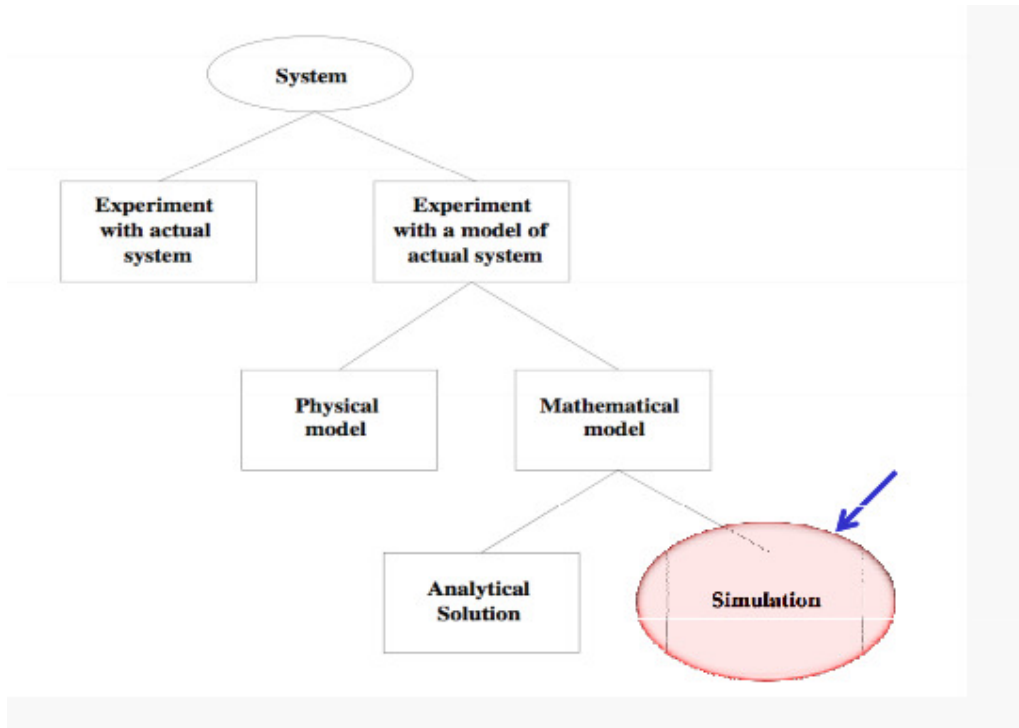


Fig 2: System Implementation and Study

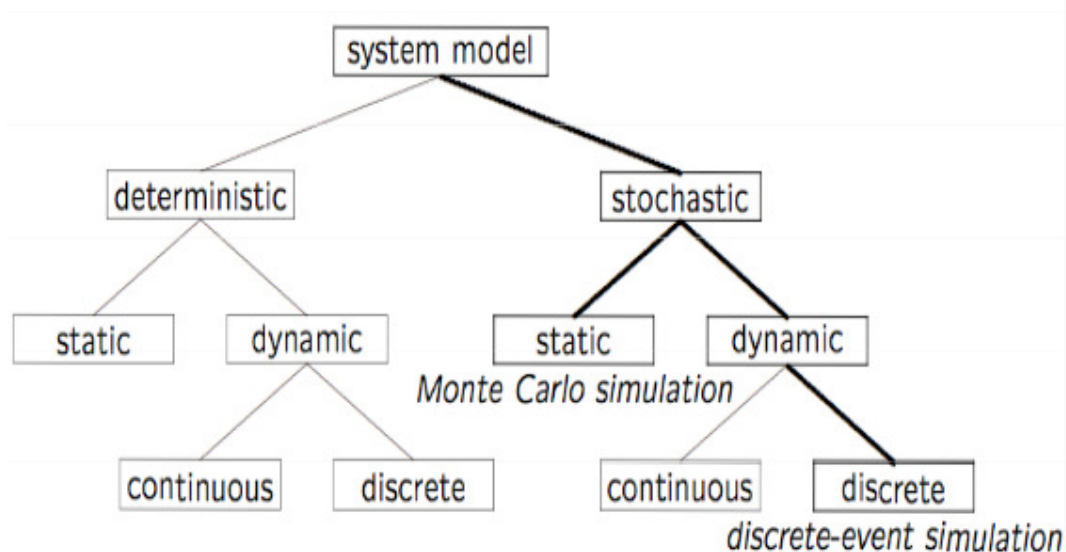


Fig 3: Model Taxonomy

Why Simulation?

- Many systems are highly complex, precluding the possibility of analytical solution
- The analytical solutions are extraordinarily complex, requiring vast computing resources
- Thus, such systems should be studied by means of simulation numerically exercising the model for inputs in question to see how they affect the output measures of performance

“Simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies (within the limits imposed by a criterion or set of criteria) for the operation of a system.”

Discrete-Event Simulation (DES)

A discrete-event simulation

Models a system whose state may change only at discrete point models a system whose state may change only at discrete point in time

System:

is composed of objects called entities that have certain properties called attributes.

State:

a collection of attributes or state variables that represent the entities of the system.

Event:

an instantaneous occurrence in time that may alter the state of the system

- **Discrete-event simulation is stochastic, dynamic, and discrete**
- **Stochastic = Probabilistic**
 - Inter-arrival times and service times are random variables
 - Have cumulative distribution functions
- **Discrete = Instantaneous events are separated by intervals of time**
 - The state variables change instantaneously at separate points in time
 - The system can change at only a countable number of points in time.
 - These points in time are the ones at which an event occurs.
- **Dynamic = Changes over time**
 - Simulation clock
 - Keep track of the current value of simulated time as the simulation proceeds
 - A mechanism to advance simulated time from one value to another
 - Next-event time advance

VIVA Questions

Q1. What is Discrete Event Simulation?

Q2. What is the importance and limitations of simulation?

Q3. What is the relationship between state, system and an event?

Q4. What is the difference between deterministic and stochastic model?

Q5. What is proof of correctness? Why mathematical models are more reliable than simulations?

EXPERIMENT 2

Aim: Introduction to NS3 and its comparison with NS2.

Description:

In this lab, we will be using the Network Simulator, NS3, available from www.nsnam.org. NS3 is a powerful program, however we will only be looking at some basic features. NS3 simulations are built in C++.

Compare NS2 and NS3 on the basis of the following parameters:

1. Programming Languages
2. Memory Management
3. Packets
4. Performance
5. Simulation Output

Some Important Points about NS3:

1. NS3 is not backward compatible with NS2; it's built from the scratch to replace NS2.
2. NS3 is written in C++, Python Programming Language can be optionally used as an interface.
3. NS3 is trying to solve problems present in NS2.
4. There is very limited number of contributed codes made with NS3 compared to NS2
5. In NS2, bi-language system makes debugging complex (C++/Tcl), but for NS3 only knowledge of C++ is enough (single-language architecture is more robust in the long term).
6. NS3 has an emulation mode, which allows for the integration with real networks.

VIVA Questions

Q1. How the scripting languages are different from Programming languages?

Q2. State the importance of NS3 over NS2?

Q3. Why it is debated to work on NS2 for research works instead of NS3 for research works?

Q4. What are the parameters on which the NS2 can be differentiated with the NS3?

EXPERIMENT 3

Aim: Install NS3 on Linux

Description:

Following are the basic steps which must be followed for installing NS3

1. Install prerequisite packages
2. Download ns3 codes
3. Build ns3
4. Validate ns3

Prerequisite packages for Linux are as follows:

1. Minimal requirements for Python: gcc g++ python
2. Debugging and GNU Scientific Library (GSL) support: gdbpython-dev, valgrind
gsl-bin, libgsl0-dev, libgsl0ldbl, Network Simulation Cradle (nsc): flex, bison
3. Reading pcap packet traces: tcpdump
4. Database support for statistics framework: sqlite, sqlite3
5. XML-based version of the config store: libxml2
6. A GTK-based configuration system: libgtk2.0-0
7. Experimental with virtual machines and ns-3: vtun, lxc

Detail steps are as follows:

1. \$sudo apt-get update / dnf update
2. \$sudo apt-get upgrade / dnf upgrade
3. Once ubuntu/fedora is installed run following command opening the terminal(ctrl+alt+T) window.
4. To install prerequisites dependency packages- Type the following command in terminal window.
5. \$sudo apt-get/ dnf install gcc g++ python python-dev mercurial bzip2 gdb valgrind gsl-bin libgsl0-dev libgsl0ldbl flex bison tcpdump sqlite sqlite3 libsqlite3-dev libxml2 libxml2-dev libgtk2.0-0 libgtk2.0-dev uncrustify doxygen graphviz imagemagick texlive texlive-

latex-extra texlive-generic-extra texlive-generic-recommended texinfo dia texlive texlive-latex-extra texlive-extra-utils texlive-generic-recommended texi2html python-pygraphviz python-kiwi python-pygoocanvas libgoocanvas-dev python-pygccxml

6. After downloading NS3 on the drive, extract all the files in the NS3 folder, which you have created.
7. Then you can find build.py along with other files in NS3 folder.
Then to build the examples in ns-3 run :
\$./build.py --enable-examples --enable-tests
If the build is successful then it will give output
"Build finished successfully".
8. Now run the following command on the terminal window to configure with waf (build tool)
\$./waf -d debug --enable-examples --enable-tests configure
To build with waf (optional)
\$./waf
9. To test everything all right run the following command on the terminal window,
\$./test.py
If the tests are ok the installation is done
10. Now after installing ns3 and testing it run some programs first to be ns3 user:
make sure you are in directory where waf script is available then run

VIVA Questions

Q1 What protocols does ns support?

Q2. How should one can start doing something (like implementing a new protocol or trying an experiment)?

Q3. What is waf and its importance in the simulation process?

Q4. How we can test that NS3 has been installed properly?

Q5. Why prerequisite packages are needed before the installation of NS3?