MODELING, SIMULATION, AND PERFORMANCE EVALUATION OF TELECOMMUNICATION NETWORKS

By

Alice S. L. Kwok

A Thesis
Submitted to the Faculty of Graduate Studies
In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Manitoba, CANADA

0-612-41729-8

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

MODELING, SIMULATION, AND PERFORMANCE EVALUATION OF TELECOMMUNICATION NETWORKS

BY

ALICE S. L. KWOK

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree

of

MASTER OF SCIENCE

ABSTRACT

Communication technology has been increasingly advancing during the last decade. Network resources are highly demanded by both the users and carriers. Demands on network resources place a potential burden on network performance. This thesis uses object-oriented modeling techniques to abstract the problem domain of a network for analysis. Commercial tools, namely COMNET III and OPNET, were used to build and to simulate the network models. The tools themselves are validated using a Poisson Process. Analytical results are used to help validate the simulations. The thesis also provides a brief description of the telecommunication standards that the models represent, allowing readers a better understanding of the network structure and behaviour.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. R.D. McLeod for giving me the opportunity to do research in this interesting area which now is in high demand in both industry and academia. I would like to thank TRLabs for providing an excellent research environment and access to equipment that normally would not have been accessible.

I would like to give special thanks to my husband J. A. Rueda for his support and encouragement during the prolonged writing period. I would also like to thank my family (mother Fei Kam, father Siu Chuen, and my siblings) for all their care and support in my studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

The main contribution of this thesis is a study of modeling and simulation techniques for telecommunication networks and a study of congestion in available bit rate type Asynchronous Transfer Mode (ATM) connections. Two commercially available tools are used throughout the thesis, namely COMNET III and OPNET. The thesis provides the theoretical background for each network model being built and compares simulation and theoretical results. The simulation tools are evaluated and recommendations made about the application of the tools in telecommunications.

1.1 The Problem

Nowadays, telecommunication technologies are growing at the same pace as computer networks. The telephone is no longer the only real-time communication equipment. The advent of new communication technologies make services such as video-conferencing possible. For example, the use of broadband communication technologies such as ATM and fiber optics allow the digital transmission of computer-generated data, in addition to digitized analog information like video and voice [1]. Present networks are able to provide wide access to a variety of information services.

There is a drastic demand for increased data transfer rates over the data communication networks. The drive behind this is the lower cost of the services and increased computer processing power. With the increasing demand of multimedia applications, the present communication systems are facing problems such as congestion [4]. The demand for internetworking is also increasing and the problem of building more robust and effective networks is becoming more visible. However, before a network is physically constructed, the planners have to determine the capacity required, based on an expected and anticipated traffic load. Computer simulation has become a viable approach to network planning.

Simulation can also be useful for planning new services or new technologies in an existing network. A simulation model can be built to approximate a physical network. With a valid model, one can easily add new services or technologies and study the impact of these new components on the network. There is no doubt that simulation is a widely accepted approach, but how well the simulation tools represent the network is another question. This thesis will compare two commercially available software programs and compare their features as simulation tools.

1.2 Scope

The thesis contains three parts. The first part introduces the technologies and dynamics of communication networks to find out how a communication network can be approximated by a model. The approximated network model enables us to perform simulations and to characterize the network traffic. The modeling technique is based on queuing theory, specifically the Markov Chain Model. The examples are the M/M/1 model and a packet-switched ALOHA network. The methods to analyze the models are both analytical and based upon network simulation. The second part of the thesis focuses on large-scale networks. A brief introduction to Integrated Services Digital Network (ISDN) and its broadband counterpart Broadband Integrated Services Digital Network (BISDN) is introduced. The objective of this part is to characterize the network performance using a Fuzzy Neural Network. In the third part, modeling and simulation of the ATM layer is presented. The modeling shows how a large physical network can be simplified into nodes, links and network traffic process. The simulation provides analyzable data, so that we can characterize network performance and predict how the network reacts under different conditions.

Chapter 2 Modeling and Simulation

In the context of telecommunication networks, modeling and simulation are used to study performance. This chapter illustrates how to model and simulate a simple network, and how to measure performance.

Section 2.1 describes modeling techniques to represent a network. Section 2.2 introduces the software tools that will be used for simulation followed by Section 2.3 which deals with performance measures.

2.1 Modeling

Networking technology is being deployed to provide wide access to a variety of information services with an unprecedented degree of convenience and flexibility. The recent advances in networking technologies have enabled a whole new set of operations. With the increase in network capacity comes an increase in the products and services that network providers can offer, and the overall system becomes extraordinarily complex. A network model is a representation of a network under study. The model can capture the physical, logical, and functional aspects of a network [5]. The model allows the user to manipulate its parameters and to observe the behaviour in different circumstances. A well constructed model has the ability to abstract and to mimic some behaviour of the original network. Usually, a model is much less complex than the underlying network, hence it is easier and cheaper to study. To properly model a communication network for simulation, we need to classify its structure and behaviour.

2.1.1 Object-Oriented Modeling

Object-oriented modeling (OOM) provides a framework in which large and complex systems can be organized in terms of their structure and their behaviour. Object-orientation is a modeling paradigm which imposes an order on raw facts so that they can be meaningfully represented for analysis. Like other modeling formalisms, OOM allows us to organize knowledge by making a classification in terms of objects, their differences, similarities, and transitions. Object-oriented techniques have the ability to capture the essence of both structure and function in the single abstraction of an object. OOM techniques utilize Object-Oriented Analysis (OOA) to allow us to formalize an object such that it can describe both aspects [7].

Object-Oriented Analysis comprises abstraction, encapsulation and inheritance. Abstraction is a mechanism to extract the important features of the target object and include it in the object model. The process defines a conceptual boundary called an abstraction barrier. Inside this boundary are the essential characteristics of the object called properties. Encapsulation consists of identifying the internal implementation details of a network element and separating those from its externally visible behaviour. Encapsulation suggests that, from the network modeling perspective, the internal details of how a network element is constructed are irrelevant, as long as its external behaviour is predictable. The classification process categorizes the objects with similar properties into an object class. The common properties of the objects may be structural, behavioural, or both.

2.1.2 Communication Network Models

Object-oriented network modeling allows one to understand the network in terms of its component objects, their attributes, and their relationships with each other. The system must first be understood and described in an abstract form before building the model.

In the case of stochastic digital communication networks, the statistical network behaviour can be obtained by developing a performance model, and obtaining the performance measures. The modeling techniques for digitized networks are based on a discrete-time approach, and the basic time unit reflects the time slots of the physical system. The reason is simply because digitized networks operate on the basis of time-slotting [5] [7]. These networks inherently lead themselves to representation with discrete-time models by specifying one-to-one correspondence between a time-unit in the model and a time-slot in the physical network.

The discrete-time theory for network modeling and analysis of the networks involves a gradual progression, starting with probability theory, then moving on through discrete-time Markov Chains and discrete-time queues, to discrete-time queueing networks. In modeling and analyzing the performance of a network, queueing theory of discrete-time queues is the main theoretical modeling tool.

There are six components describing the queueing system, namely arrival process, service process, number of servers, waiting room, customer population and service discipline.

1. The arrival process is a stochastic process describing how customers, usually messages in a communications system, arrive into the system. Very often people use interarrival information to describe the process. Please see the following subsection for more detail on the interarrival time.

2. The service process is a stochastic process describing the length of time a server is occupied by a customer. For example, the length of time a communication channel takes to transmit a message in a computer network.

3. The number of servers in the system, such as a node in a communication network.

4. Waiting room is the limit to the number of customers that can be accommodated to wait for service, including those currently being serviced.

5. Customer population is the limit to the total number of customers, including potential ones. This can be considered as part of the arrival process, since the number of potential customers can change the rate of arrivals.

6. Service discipline is the rules that decide which customer or customers in the queue to serve.

The gradual progress method can be used to construct a very accurate network and to reproduce the stochastic behaviour of the network, but very often a simple good approximation is able to provide the desired solution as well. Nonetheless, as the time unit in the discrete event tends to zero, the result is a continuous-time system.

2.1.3 The Analytical Method

2.1.3.1 Modeling Stochastic Processes

Stochastic processes are defined as random processes that change with time [49]. We can define the stochastic process as a set of random variables $\{x_t, t \in T\}$ defined on the same sample space T. The set T is usually interpreted as a set of time instants. When t is countable we have a discrete-time stochastic process, and when the elements of T are a continuous set of real numbers, we then have a continuous-time stochastic process. $\{x_t\}$ is called the state of the process at time t. A stochastic process is said to be strictly stationary if the distribution of (x, x_{t+s}) is independent of t [30].

A Poisson process is a special case of a Markov process. A Markov process is a stochastic process such that if at a given time its state is known, then its subsequent behaviour is independent of its past history. This memoryless property makes Markov processes very popular in network modeling. For example, the interarrival time is the time between two successive events in an arrival process that can represent the time between messages generated by an end-user. In continuous-time systems, interarrival times are exponentially distributed so that the arrival process is Poisson. In a discrete-time system, interarrival times are geometrically or binomially distributed. These distributions also apply to the service process which process the message generated by the end-user.

2.1.3.2 Poisson Processes

Poisson processes are frequently used to define the arrival process of information units in a communication or computer network. As the name implies, it is based on the Poisson distribution, and it is a pure random arrival process. The required conditions of a Poisson process are:

1. The initial number of customers in the system is zero.

2. There are no multiple arrivals at the same time. That is, the probability, *P*, that two or more events, *N*, occurred in a given instant of time, *h*, is zero. If the rate for the event to occur is λ, the requirement can be written as

$$P(N(h)=1)=\lambda h+o(h) \text{ and } P(N(h)\geq 2)=o(h). \tag{2. 1}$$

3. The number of events, n_1, that occurred in a time interval, t_1-t_0, is independent of the number of events, n_2, that occurred in another interval, t_2-t_1, providing that the two intervals are disjointed.

$$P(N(t_1)-N(t_0)=n_1, N(t_2)-N(t_1)=n_2)=P(N(t_1)-N(t_0)=n_1)P(N(t_2)-N(t_1)=n_2). \tag{2. 2}$$

4. The processes in the system are stationarily incremental. That is, for a given time interval (t_1,t_2), the probability for the events to occur is the same as the interval shifted to (t_1+s,t_2+s).

$$P(N(N(t_2)-N(t_1)=n)=P(N(t_2+s)-N(t_1+s)=n). \tag{2. 3}$$

If the above conditions are true for a process, then the number of events follows the Poisson distribution (see proof in Appendix A).

Property 1: *Poisson Probability Distribution Function.*

The probability that n events occur in time t can be written as;

$$P\left(N(t+s)-N(s)=n\right)=e^{-\lambda t}\frac{(\lambda t)^n}{n!}. \tag{2. 4}$$

This is the equation that can be used to calculate the probability that n packets arrive in a time period t.

2.2 Simulation Tools

Simulation here means modeling of systems and their operations using various means of representation [4]. In terms of network simulation, we are interested in characterizing the physical network and reporting its efficiency. Simulations help managers and engineers design the complex systems, and investigate and understand systems that either do not exist or cannot be used for experimentation.

One of the objectives of this thesis is to study large-scale physical networks. To simulate a large-scale network, it is desirable to model at a higher level of abstraction rather than detailed modeling. However, detailed modeling is important, because it provides a better approximation of a physical network with respect to both the network architecture and the processing mechanisms. But in most of the cases, detailed modeling is time consuming and not practical. When simulating a network, it is assumed that the simulation model is simplified and yet can still represent the network under study.

The software tools that will be used to simulate the models are COMNET III and OPNET. They are both based on OOM techniques.

As it is typical, the operation of a network is described using queuing theory. There are specific statistical functions for different kinds of processes or incoming transitions, which also apply to telecommunication networks. The nodes and links can be approximated by statistical functions because of their non-deterministic properties. However, the important parameters such as the topologies of the networks, network protocols, and the message transmission protocols should be modeled according to the physical network.

2.2.1 Comparison of COMNET III and OPNET

This section outlines some valuable features of the network modeling and simulation software. The first tool that will be described is COMNET III followed by OPNET.

2.2.1.1 COMNET III

COMNET III is a discrete event simulation tool for modeling computer and communication networks at a very high level of abstraction. COMNET III uses OOM techniques to allow users to build their models in a drag-and-drop fashion. COMNET III is very user friendly, all the network configurations are done graphically through the Graphical User Interface (GUI), or alternatively, the user can import the current topology from a Network Management System. There is no programming required in building a network model. The objects of the network are represented by icons, namely clouds, subnetworks, network topologies, nodes, and links.

The designer or network planner has the choice of using a hierarchical approach to organize the network layout, by building portions of the network in subnetworks. The tool has the ability to trace, to animate the simulating model, to interact with the model while performing simulation, and to display the status of packets on a dynamic graph. If the animation is turned on, an animated picture will be displayed on the window showing how and where the packets are generated and transited in the model.

COMNET III can run multiple, independent replicas of a simulation and generate a statistical report for each one, including performance measures. The software will automatically detect possible errors before every simulation run. Alternatively, the designer can also use the built-in network validation tool to validate the network model before simulation. COMNET III includes an object library composed of a variety of pre-defined network components. The compiling and linking procedures are all done in a way transparent to the designer. The parameters of these components can be easily adjusted through the specification forms to match real-world objects. However, the normal version does not allow users to define their own functions to be used in the simulation.

COMNET III does not support mobile communication systems which are the main component of personal communication systems (PCS). On the other hand, satellite communication systems and any other wireless networks can be modeled, as long as the network is considered to be geo-stationary. In that case, the network can be approximated in terms of propagation delay and network contention.

COMNET III models are portable. The same model built on the SUN version can be imported to the DEC version.

2.2.1.2 OPNET

OPNET also uses OOM techniques to build the models, but at a lower level of abstraction. The models are specified in terms of objects, each with configurable sets of attributes. The package supports flexible definitions of new objects with programmable characteristics and behaviour in order to address as wide a scope of systems as possible.

OPNET is a specialized tool for communication networks and information systems. Its model building blocks focus primarily on communications and information processing to accelerate development efforts for networks and distributed systems. The hierarchical approach allows designers to build their networks in parallel with the structure of actual communication networks. The level of detail can range from processing units to Wide Area Networks (WANs).

OPNET model specification is done graphically in combination with the necessary programming. The graphical editors assist the modeling process through its graphical specifications. These editors provide an intuitive mapping from the modeled system to the OPNET model specification. However, the designer will spend most of the time defining the functions and OPNET function calls.

OPNET provides a flexible, high-level programming language for modeling communications and distributed systems with actual protocols and algorithms. The language is based on the C programming language with a very large library of functions. The library provides, for example, the capability of assigning a uniformly distributed integer to a variable. If we want to randomly assign to a variable called *address_dist* an integer from 0 to 3, we can use the function *op_dist_load* as follows,

$$address\_dist = op\_dist\_load(\text{"uniform\_int"}, 0,3). \tag{2. 5}$$

The above command uniformly assigns an integer from 0 to 3 to the variable *address_dist*.

After the network model is built, the model has to be compiled and linked with all the associated objects in order to perform a simulation run. Model specifications are automatically compiled into executable, discrete-event simulations implemented in the C programming language.

- OPNET includes a tool for graphical presentation and processing of simulation output. The tool also comes with a set of filters to let the analyst generate a different form of statistical results to assist data analysis.
- Debugging tool. All OPNET simulations automatically incorporate support for analysis via an interactive debug software. The debugging software cannot be accessed graphically, so programmers have to use the command lines. To run the debugging tool, all the programmer has to do is to run the simulation file through the UNIX command line, but in a debug model (by adding extra "-debug" option). When the programmer is in the debug mode, he/she can jump to a particular point of the simulation and start the trace process. The programmer can also set different levels of information to be displayed during trace. The user is allowed to step-over the simulation process or set a break point to jump-over the simulation, and quit the debugging simulation process any time.

The animation support is not as easily accessible as in COMNET III. All animation has to be done by creating an animation probe at the Probe tool which is the recording object used to record the animation history at various levels. Extensive support for developing custom animation is also provided. However, there is an unknown problem with the tool. It is possible to encounter errors during recording. For some unknown reason the recorded animation history data indicate that the data are opening unspecified windows with 0X0 dimensions continuously.

2.2.1.3 OPNET TOOL SET

OPNET is an aggregated software tool. It is composed of eight independent software tools, namely; Network Editor, Node Editor, Process Editor, Parameter Editor, Probe Editor, Filter Editor, Simulation Tool, and Analysis Tool.

The Network Editor is used to specify the topology of a communication network model in terms of the objects that it contains and the relationships between them. Node objects include subnetworks, fixed communication nodes, mobile communication nodes, and satellite communication nodes. Link objects that connect these node objects are the simplex point-to-point link, duplex point-to-point link, bus link, and tap. Tap is used to attach fixed communication nodes to a bus link so that they may transmit and receive packets.

The Node Editor is used to specify the structure of device models that can be instantiated as nodes in the Network Editor. Nodes are composed of several different types of objects called modules. Modules are connected together to describe the flow of data within the node model. The module objects are the processor, queue, ideal generator (to generate traffic), clocked generator (which is the same as ideal generator but packets are generated at the discrete points in time), point-to-point transmitter, point-to-point receiver, bus transmitter, bus receiver, radio transmitter, radio receiver, and antenna. The connections between modules are done by packet stream objects (to convey packets) and statistic wires (wires that monitor the state of another module).

The Process Editor is the environment in which Proto-C specifications are developed. Proto-C is a mixed graphical and textual language specifically designed to support model development for interrupt-driven processes such as communication protocols and distributed algorithms. Proto-C process models are represented graphically as state transition diagrams (STD) that consist of state and transition objects, and blocks of code embedded within its states.

The Parameter Editor is actually not an editor itself, but an environment that encompasses several small editors. Each of these editors is used to specify a single type of model. These parameter model types are the packet format, interface control information format, probability density function, modulation, and antenna pattern.

The Probe Editor is used to define data collection requirements prior to executing a simulation. Creating an object called a probe enables each desired output.

OPNET simulations are UNIX programs that can be executed in the normal fashion from an ordinary shell command line. OPNET simulations are capable of generating two types of data files, namely, output vector and output scalar files. The Simulation Tool allows any number of separate simulation runs of various models to be specified in a tabular format.

2.3 *Performance Measures*

The objectives of modeling and simulating a network is to determine how well the network performs. There are many important measures that can be used for assessing the performance of a communication or computer network. Among these are the throughput, average message delay, probability of buffer overflow, probability of message loss, reliability, and adaptability to variation in traffic [30]. The definitions for the more popular performance measures are as follows:

1. Throughput is defined as the number of successfully transmitted messages per mean transmission time of a message.
2. Message delay is defined as the time interval, in units of the average transmission time of a message, from the moment a message is generated to the instant it is correctly received.
3. Probability of buffer overflow is defined as the fraction of messages that are lost due to no buffer space being available at the time of their arrival. This measure is important in simulating certain types of traffic such as voice and video.
4. Reliability is defined as mean time to fail. This measure is important, for example, in defining the links in the network.

2.4 Modeling a Simple Network

2.4.1 The M/M/1 Markov Chain Modeling of Data Transfer

The objective of the example is to demonstrate how to model a physical network using Markov Chain queueing theory, and how to use the model to estimate the network performance. The continuous-time Markov Chain model will be used.

The basic element in the communication and computer network is the transferring of the data traffic. For example, a Local Area Network (LAN) consists of N workstations and a router as shown in Figure 2.1, where N is any positive integer. The router checks the destination address of a packet and directs it to the appropriate path. In order to apply a Markov Chain Model, the network must satisfy the following conditions:

- The workstations are randomly generating fixed sized packets at rate λ_i [packet/second] following an exponential density,

$$P_i(t) = \lambda_i e^{-\lambda_i t}, \tag{2. 6}$$

  where $i = 1, 2, .., N$ labels the workstations.

- The router processes and transfers the packets at μ [packets/second] following an exponential distribution service rate as well,

$$P(t) = \mu e^{-\mu t} \text{ and } \mu > \sum_{i=1}^{N} \lambda_i . \tag{2. 7}$$

- The router has an input buffer with infinite length of queue size, and with FIFO as service discipline.

The goal is to find the delay of the packets and the size of the router's input buffer.

2.4.2 The M/M/1 Model

The M/M/1 model of the network has N input sources. The sources here are representing the workstations where the traffic is originated. The interarrival of the packets from each source follows an exponential probability distribution function at rate λ_i. The processor node is passing the packets at rate μ with an exponential probability distribution function as well. The circles connected to the input buffer of the router symbolize the N workstations.

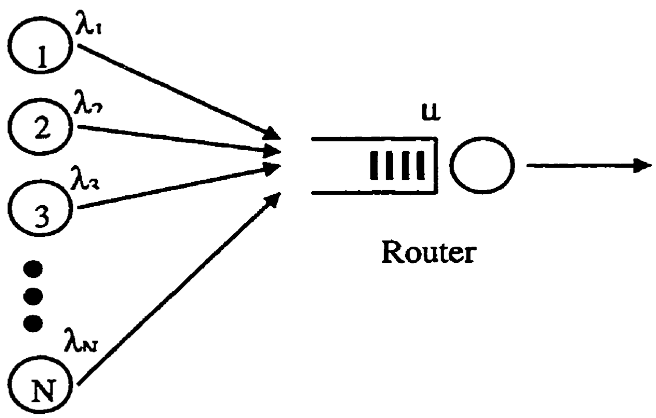


Figure 2.1: M/M/1 model of the LAN.

Let us assume that the network satisfies all the conditions of Poisson Process as stated before, so that the problem can be modeled as an M/M/1 Markov Chain. The model allows us to estimate how long a packet will take to pass through the router and the size of the queue at the buffer when the system is in steady state. The delay of the packet at the router can be interpreted as the waiting time W of the packets at the router, and the buffer size has to be greater than the mean number of packets L at the router multiplied by the packet size.

The first step to solve the problem is to simplify the problem by reducing the number of input nodes based on the following superposition property (see the proof in Appendix A):

Property 2: *Superposition property.*

The total packet injection rate to the router can be written as $\lambda = \sum_{i=1}^{N} \lambda_i$.

2.4.3 Analytical Study

After simplifying the interarrival rate, the M/M/1 model for a LAN can be described by

$$\mathbf{P(t)Q} = \frac{d}{dt}\mathbf{P(t)}, \tag{2. 8}$$

where $\mathbf{P(t)} = [p_{ij}(t)]$ is the probability distribution function matrix and $\mathbf{Q} = [q_{ij}(t) = q_{ij}]$ is the transition rate matrix derived from Kolmogorov's Backward Equation. The transition rate diagram of the system is illustrated in the following diagram.

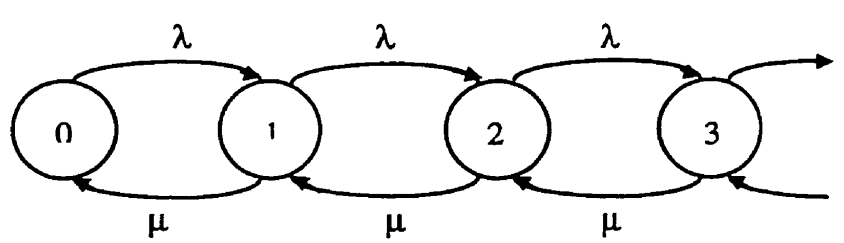


Figure 2.2: Transition rate diagram.

When the system is in equilibrium, that is when the probability distribution function converge to steady-state probability and the probability of a certain number of packets at the queue stays constant, the system is expressed by, $\mathbf{P(t)} = \Pi = [\pi_i]$. The symbol Π is the steady-state probability matrix. The rate of change of the probability will be zero as t approaches to infinity. The equations will become,

$$\lim_{t\to\infty}\frac{d}{dt}\mathbf{P(t)} = \mathbf{0} \qquad \text{and} \qquad \lim_{t\to\infty}\mathbf{P(t)Q} = \Pi\mathbf{Q} = \mathbf{0}. \tag{2. 9}$$

The transition rate matrix of the M/M/1 model is

$$\mathbf{Q} = \begin{bmatrix} -\lambda & \lambda & 0 & 0 & 0 & \cdots \\ \mu & -\mu-\lambda & \lambda & 0 & 0 & \cdots \\ 0 & \mu & -\mu-\lambda & \lambda & 0 & \\ 0 & 0 & \mu & -\mu-\lambda & \lambda & \\ 0 & 0 & 0 & \mu & \ddots & \\ \vdots & \vdots & & & & \end{bmatrix}. \tag{2. 10}$$

By substituting the transition rate matrix into the steady-state equation, we can obtain a set of equations.

$$\begin{aligned} &\lambda\pi_0 - \lambda\pi_1 = 0 \\ &\lambda\pi_0 - (\mu+\lambda)\pi_1 + \mu\pi_2 = 0 \\ &\lambda\pi_1 - (\mu+\lambda)\pi_2 + \mu\pi_3 = 0 \\ &\vdots \end{aligned} \tag{2. 11}$$

Solving the equations, the result becomes

$$\pi_0\left(1+\frac{\lambda}{\mu}+\left(\frac{\lambda}{\mu}\right)^2+\left(\frac{\lambda}{\mu}\right)^3+\left(\frac{\lambda}{\mu}\right)^4\cdots\right)=1. \tag{2. 12}$$

The sum inside the brackets is a geometric series. Note that the system will diverge and become unstable when $\frac{\lambda}{\mu} \geq 1$. This is the reason why $\mu > \sum_{i=1}^{N} \lambda_i$ is assumed in the model.

The steady-state probability for no packet in the system is given by the following equation

$$\pi_0 = \frac{1}{1+\sum_{k=1}^{\infty}\left(\frac{\lambda}{\mu}\right)^k} = 1-\frac{\lambda}{\mu}. \tag{2. 13}$$

The steady-state probability for k packets in the systems where $k>0$ is

$$\pi_k = \left(1-\frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^k. \tag{2. 14}$$

After finding Π, we can use the generation function that is the Z-transform of the sum of the steady-state probabilities.

$$\begin{aligned}\Pi(Z) &= \sum_{k=0}^{\infty} \pi_k Z \\ &= \left(1-\frac{\lambda}{\mu}\right)\sum_{k=0}^{\infty}\left(\frac{\lambda}{\mu}\right)^k \\ &= \frac{1-\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}Z}.\end{aligned} \tag{2. 15}$$

The first moment of the generation function at $Z=1$ is the average number of packets N in the system. Next, we apply Little's result, which states that the average delay in the system is $T=N/\lambda$, and obtain the following expression

$$N = \frac{d}{dz}\Pi(Z)_{z=1} = \frac{\lambda}{\mu - \lambda}$$
$$T = \frac{1}{\mu - \lambda}. \tag{2. 16}$$

For an M/M/1 queue, the average number of packets C in the service (being processed by the router) is

$$C = \sum_{k=1}^{\infty} \pi_k = 1 - \pi_0 = \frac{\lambda}{\mu}. \tag{2. 17}$$

Therefore, the number of customers L in the router queue is

$$L = N - C = \frac{\lambda^2}{\mu(\mu - \lambda)}. \tag{2. 18}$$

Similarly, the average waiting time of the packets at the router queue is the average time delay minus the average time spent in the routing process. Since the processing time is exponentially distributed with rate μ, then the average processing time is $1/\mu$. The waiting time at the queue is

$$W = T - \frac{1}{\mu} = \frac{\lambda}{\mu(\mu - \lambda)}. \tag{2. 19}$$

To justify the validation of the estimated result, simulation of the same model has been done.

2.4.4 Comparing the Analytical and Simulation Results

After introducing the base element in network modeling, this section provides a simple example to demonstrate how to model and simulate a network. Let us assume that a LAN has two workstations and a router. The workstations generate fixed size packets to the outside world through the router. The router is responsible for sending the packets to the right path. The interarrival processes of generating the packets are Poisson process with exponential probability distribution functions with $\lambda_1 = 5$ *packets/sec* and $\lambda_2 = 2$ *packets/sec* respectively. Both workstations are generating *1 Kbyte (8000 bits)* size packets. The service process at the router is Poisson with an exponential probability distribution function with $\mu = 58000$ *bits/sec*. The expected number of packets generated from workstation 1 is λ_1, that is 5 packets per second. The number of packets generated by workstation 2 is therefore 2 packets per second. From Equations (2.18) and (2.19), the calculated average queue length at the router buffer is 27 packets (or 27 Kbytes), and the expected waiting time per bit is 0.00048 second or 3.86 seconds per packet.

2.4.4.1 COMNET III Model

In COMNET III the LAN can be modeled as a router connected to two computer nodes, src1 and src2. The computer nodes are connected to two source nodes, Msg1 and Msg2, which generate the traffic. When the router receives packets from the source, it sends the packets to the outside world, the sink. The function of the sink is simply the final state of traversing the packets. The workstations are assigned with no packet processing delays, such that every generated packet from the source nodes will be transferred to the router instantaneously. The delay of the packet is solely the waiting time at the input buffer of the router. The faster the router services the shorter the waiting time for the packets. The router processes the packets with the predefined rate then sends the packets to the sink node. The data collected are the number of packets generated by the two source nodes, and the message delay from the source to the sink, which is also the delay at the router.

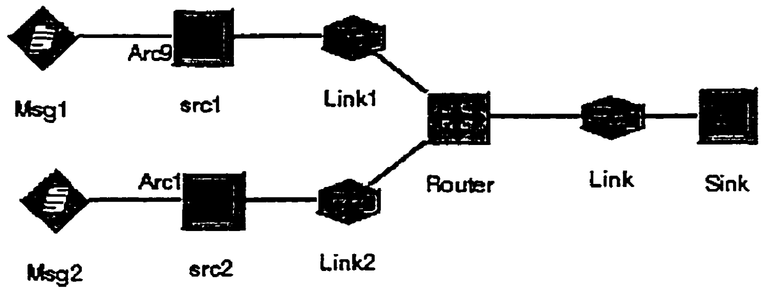


Figure 2.3: COMNET III LAN network model.

The network model was simulated five times for 10,000 seconds each. The results of the simulations are identical. At the end of a simulation, COMNET III generates a statistical report. The recorded statistics including the number of packets generated by each source, and the delay time from the source nodes to the sink. The source nodes also recorded some statistic data as shown in Figure 2.4. The figure and the simulation report indicate that the source nodes are generating packets at the expected rates. On average each source, source nodes, src1 and src2, are generating packets at 5 packets per second and 2 packets per second. The average delay time, also called the waiting time, of a packet is 19.6 seconds with standard deviation of 14.1 seconds. The average buffer size is 140,803 bytes (or 137.5 packets) with standard deviation 102,197 bytes. The maximum delay time is 50.1 seconds, and the maximum buffer size is 355,328 bytes or 347 packets. The overall performance of the network is summarized in the last section of the chapter for comparison.

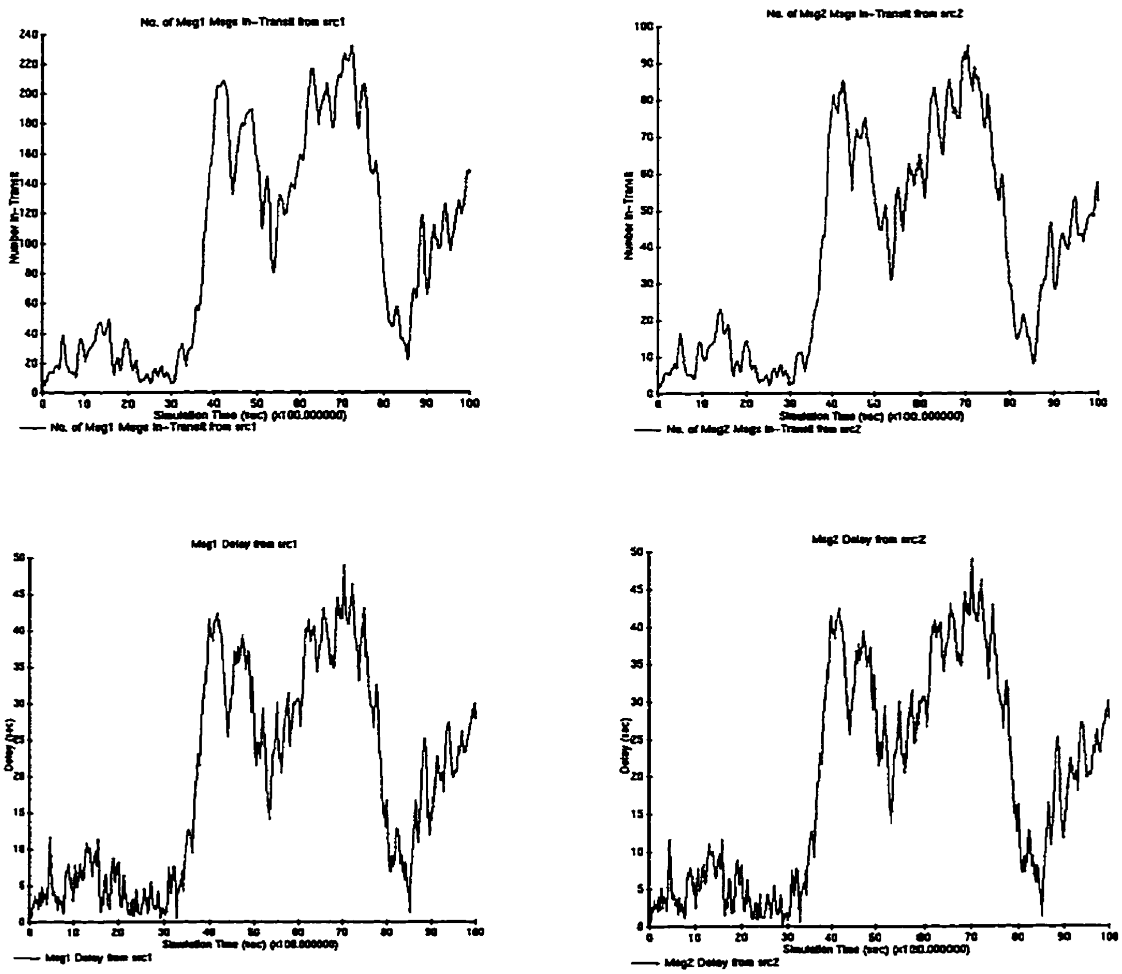


Figure 2.4: COMNET III number of packets in transit and packet delay.

2.4.4.2 OPNET Model

The model of the network in OPNET is illustrated in Figure 2.5. The ideal generators in OPNET are responsible for generating traffic, and represent the workstations of the LAN in this study. The ideal generators do not perform any processing, but simply generate packets as specified. The router is represented by its input buffer queue. The router sends the incoming packets to the outside world denoted by a sink node. Whenever the router receives a packet from either of the source nodes, the packet will be put at the input queue of the router, then sent to the sink.

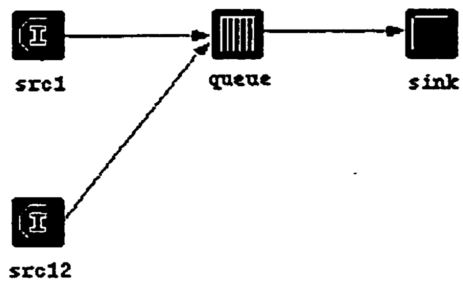


Figure 2.5: OPNET LAN network model.

The OPNET M/M/1 model was simulated for 8,000 seconds and the result is displayed in Figure 6. The data recorded includes the number of packets generated from each source, the delay time and packet size of a packet at the input queue of the router. The average number of packets generated from src1 and src2 is 5 packets per second and 2 packets per second respectively. The average waiting time at the buffer for each packet is converging to 7 seconds. The average queue length is 50 packets. The maximum delay is 12.5 seconds, and the maximum buffer size is 90 packets or 91,160 bytes. The summary of the results is included in the last section of this chapter as well.

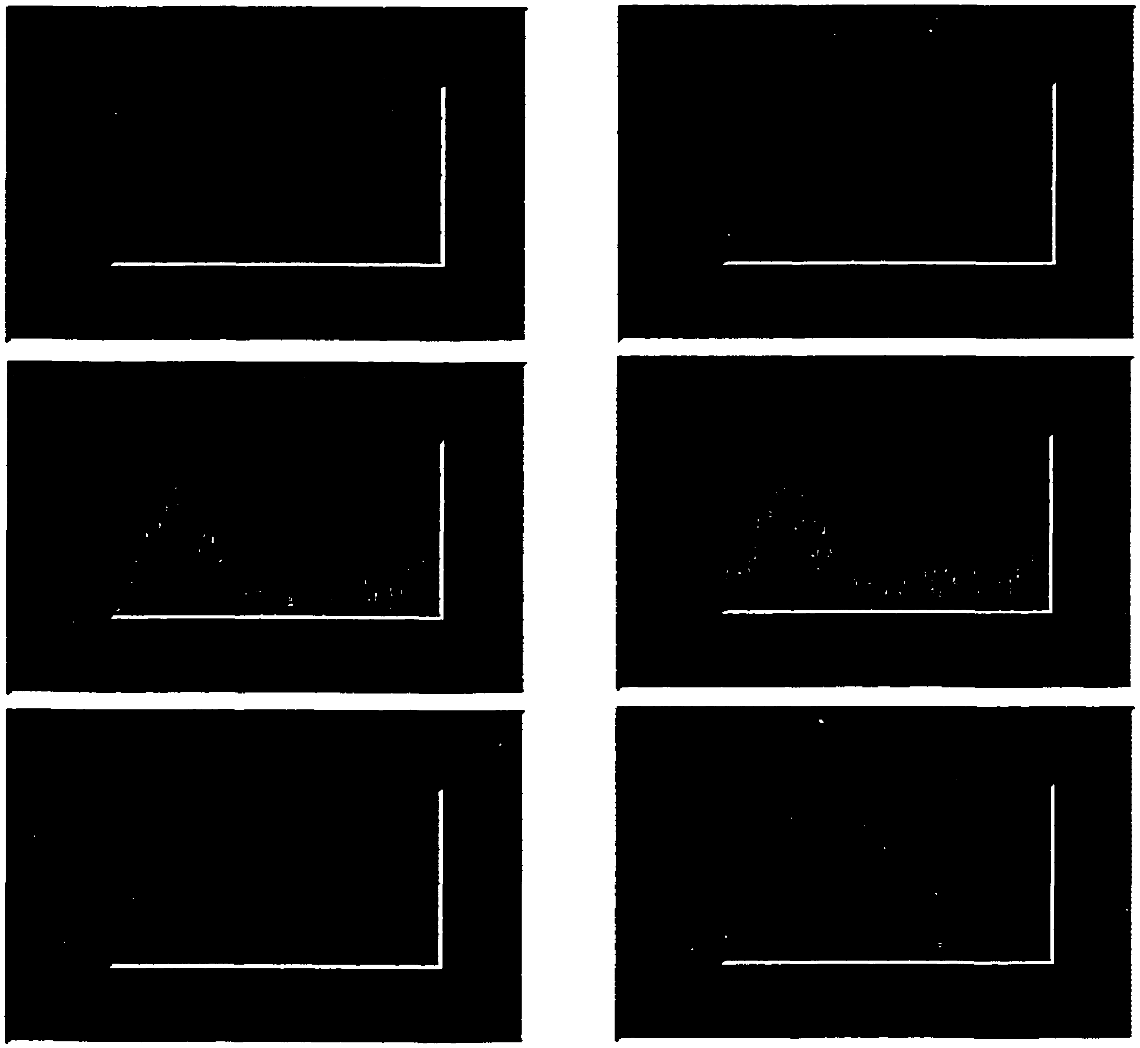


Figure 2.6: OPNET LAN packet delay and packet size.

2.4.5 Comparison of the Results

From the summary in Table 2.1, the values indicate that the simulation results are very different from the analytical calculations. It was expecting that for such a simple model the simulation results should have been much closer to the analytical results. It looks like the two simulation models have a slightly higher interarrival time than the desired value

(56000 bps). The results would have matched perfectly if the interarrival time has changed to the ones in Table 2.2.

	Analytical Calculation	COMNET III Model	OPNET Model
Average packets (src1/Msg1)	5	5	5
Average packets (src2/Msg2)	2	2	2
Average packet delay time (sec)	3.86	19.6	7
Maximum delay time (sec)	N/A	50.1	12.5
Average buffer size (KB)	27	137.5	50
Maximum buffer size (KB)	N/A	347	90

Table 2.1: Summary of the results.

Interarrival Time (bps)	56000	57000	57500	57600
Average Buffer Size (pkts)	27	56	114	143
Average packet delay time (sec)	3.86	7.86	15.86	19.8
Similar Model	Analytical	OPNET	N/A	COMNET III

Table 2.2: With different interarrival values.

Another observation from the simulation process is that COMNET III and OPNET use different methods to send packets. Analytically, one packet will be sent at a time, which is the same as in OPNET. In COMNET III the packets are sent one byte at a time. From the mean packet (pkt) flow diagrams of Figure 2.6, it is clearly visible that sometime more than one packet is generated at a time. The results of this are the differences in the maximum delay time and the maximum buffer size.

In spite of the difference, we can estimate the delay of sending a packet across the router and the time a packet will spend at the router queue. The possible delay at the router is more than a few seconds for such a busy network, and a large buffer of at least 100 Kbytes would be required.

Chapter 3 Network Layer Simulation

In this chapter, the Open Systems Interconnection (OSI) Reference Model (OSIRM) will be introduced, and it will be used as a guideline for other protocols in the following chapters. In network protocol standardization, the OSIRM is used as a reference model to introduce new protocols and services. This chapter provides a condensed summary of the OSIRM and explains the function of each layer. The first section presents the seven layers of the OSIRM. In Section 2, the method by which messages traverse between networks according to the OSIRM is covered.

3.1 The OSI Reference Model

In 1984, the International Organization for Standardization (ISO) released the OSIRM [4] [8] [10] [11] for the purpose of fulfilling the need of implementing interoperable networks. It then became the main architecture model for inter-computer communication. Subsequently, it has become the best tool for teaching network technologies. Originally, the OSIRM was dedicated for open systems such as computer or data communication networks instead of telecommunication networks. There is a difference between these two kinds of networks. In data communications, the key issue is the openness of the network, while in telecommunication networks, it is the efficiency. In spite of the difference, telecommunication networks are trying to match the network architecture with that of OSIRM.

3.1.1 The OSI Layers

OSI Reference Model is a hierarchical model with seven layers, see Figure 3.1. The last three layers are concerned with application services and are composed of the so called application-layer protocols. The lower four layers are concerned with data transport issues, and consist of the so called lower-layer protocols. The lower-layer protocols are responsible for handling networking and transmission services. Each layer of the model has a predetermined set of functions.

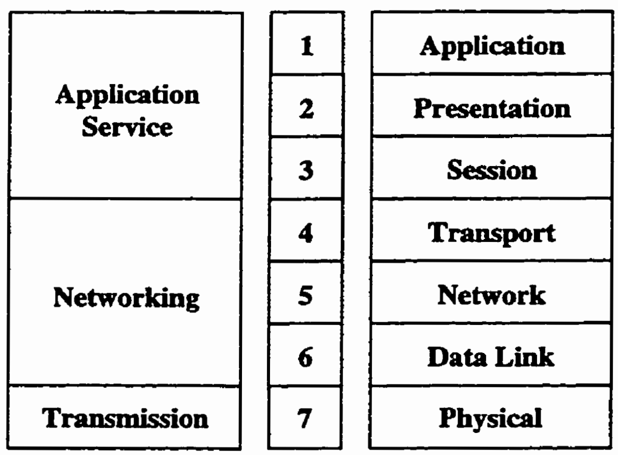


Figure 3.1: The OSI Reference Model (OSIRM).

The functions of the layers are as follows:

1. Physical Layer: The physical layer defines the electrical, mechanical, procedural, and functional specifications for activating, maintaining, and deactivating physical

connections between the data-link layer of the end systems. The objective is to provide a transparent transmission and interface technique to the physical media.

2. Data Link Layer: The data link layer provides reliable transit of data across a physical link concerning physical addressing, error detection and correction, line status notification, ordered delivery of frames, and flow control.

3. Network Layer: The network layer is where path selection is made and provides connectivity between two end systems that may be located on geographically diverse subnetworks. The network layer provides a transparent transfer of data between entities. The network service negotiates QoS with transport entities at the time of establishment of a network connection.

4. Transport Layer: This is the boundary of the lower-layer protocols. Its function is to provide a reliable transport service over an internetwork. The transport layer provides mechanisms for the establishment, maintenance and orderly termination of flow control, so that the applications can obtain a consistent view of the lower layers.

5. Session Layer: The session layer establishes, manages, and terminates sessions between applications. In addition to the basic regulation of sessions, the session layer offers provisions for data expedition, class of service, and exception reporting of session-layer, presentation layer, and application-layer problems.

6. Presentation Layer: The presentation layer ensures that information sent by the application layer of one system will be readable by the application layer of the

other system. If necessary, the presentation layer translates between multiple data formats by using a common data representation format.

7. Application Layer: The application layer only provides services to the application processes outside the scope of the OSI model. It provides a means for application programs to access the OSI environment. This layer contains management functions and generally useful mechanisms to support distributed applications. In addition, general purpose applications such as file transfer, electronic mail, and terminal access to remote computers belong to this layer.

3.2 Transporting Messages through Networks

When messages are passing between networks, they are segmented and reassembled, often into different formats [6] [11]. The general name for these formats is Protocol Data Unit (PDU). The logically grouped units (terms) of information are frames (in layer 2), datagrams (in layer 3 as in Internet Protocol), segments (in layer 4 as in Transmission Control Protocol), and other formats like cell and packet. They all represent the messages of the network, but each of them is specified for a certain use. A frame denotes an information unit whose source and destination is a link-layer entity. A packet denotes an information unit whose source and destination is a network-layer entity.

OSI communication between two end systems, which are often called hosts, can be interpreted as communication carried out between the adjacent layers at the sender and receiver nodes. When the message is received by the physical layer, it is transmitted through the link to the receiver node through the intermediate systems or relays.

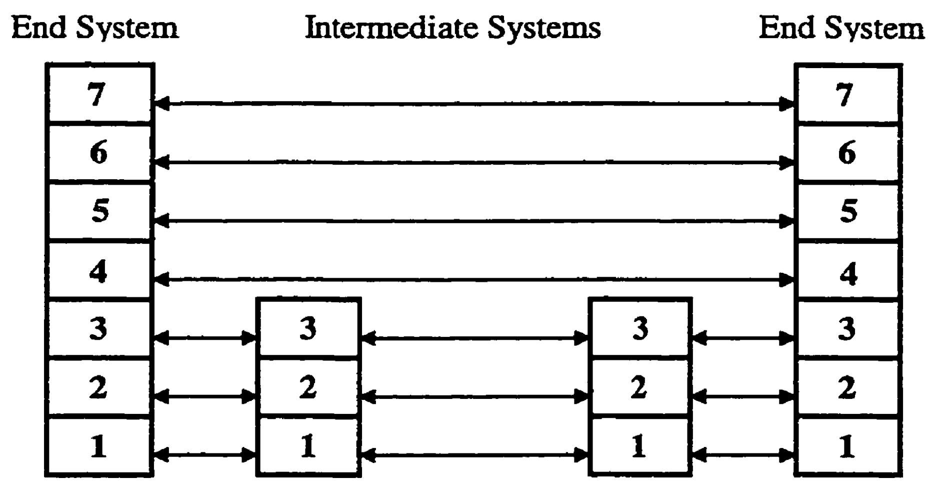


Figure 3.2: Communication between nodes.

3.2.1 Internetworking

From time to time there are needs for communications between LANs. This includes demand for communications capabilities between LANs acquired from different vendors and located at great distance from each other.

In talking about interconnecting LANs, three different levels are commonly used: local area networks (LANs), metropolitan area networks (MANs), and wide area networks (WANs). They can be differentiated by the maximum distance between terminals connected to the networks. The distance can be a few kilometers for LANs, 80 km for MANs, and up to the entire earth for WAN.

3.2.2 Intermediate Systems

The intermediate system is an abstract term used to identify the entities of the OSI environment which perform only the functions of the lower three layers. An intermediate system can be a real subnetwork or it may be a unit used to interconnect one or more real subnetworks. These units may be simple repeaters (at the physical layer), bridges between local area networks (at the datalink layer), routers (at the network layer) between different networks, or application layer gateways [33].

3.2.2.1 Gateways

A gateway is where information is converted by an application for further conveyance to another application. This simply is the full processing of seven layers of communication functionality. It is most applicable to store-and-forward types of applications or to conversion between totally different architectures.

3.2.2.2 Router

A router is a term applied to a type of internetworking unit (IWU). IWU is an actual piece of equipment which behaves as an intermediate system and is used to interconnect multiple subnetworks. An IWU is a relay system which performs its relaying function at the network layer. This relaying function can be viewed as a set of procedures by which a system forwards data from one system to another.

The IWU in the router uses a protocol in common with end systems, independent of any subnetwork specific protocol. The basic principle in applying this term is the IWU only performs routing and no "relaying" functionality from one subnetwork to other [20].

3.2.2.3 Relays

A relay does not operate as a source or a sink of data, but maintains consistency of the semantics of the incoming and outgoing data streams. Bridges and repeaters are relays.

A bridge is a relay that operates at layer 2 within a LAN environment. More specifically, it operates as a relay with respect to the layer 2 portion of the Medium Access Control (MAC).

A repeater is a relay that operates at layer 1 of the OSIRM. Its purpose is to convey a transmission signal in environments where physical limitations would otherwise prevent its further propagation. For example, IEEE 802 LANs have a limitation on the physical distance of the coaxial cable used for interconnection.

3.3 Modeling and Simulation of a Satellite Network

This section uses a satellite network as an example to demonstrate how COMNET III and OPNET can be used to model the Network Layer of the OSIRM with a router node (a hub) to transmit the data to the appointed destination.

A geostationary communication satellite itself is a relay station in outer space. The satellites are placed on an orbit 22,300 miles above the earth traveling at the same angular velocity as the rotation of the Earth. Therefore, a satellite is stationary with respect to the surface of the earth. One satellite can cover 40 percent of the earth surface, but also induces a delay of 0.239 seconds for signals to complete a round trip between a hub and a satellite.

In spite of the delay introduced due to the distance, and assuming that there is no delay at the hub, the satellite network works similarly to an M/D/1 model, where D is determined by the bandwidth of the channels. In this example, eight geo-stationary satellites are sending packets to one another through the central hub. The hub relays the received messages to the destination satellite. The bandwidth of the channels is 9600 bits-per-second (bps) and the satellites are generating traffic randomly according to an exponential distribution for the interarrival times. The packet size is fixed 9600 bits.

3.3.1 The M/D/1 Model

The M/D/1 model is a special case of the M/G/1 model which has Poisson arrivals, general service time distribution, and a single server. The process time of the M/D/1 model is deterministic.

In the analytical model, we can neglect the time delay introduced by the link propagation from the ground to the satellite. The link delay, which can always be added to the calculated result at the end, is not of interest in this model.

The equivalent model for the satellite network consists of eight identical sources with the same traffic interarrival time, and a satellite hub with a deterministic process rate that is equal to the bandwidth of all the channels, μ=9600bps. Since this model assumes that the arrival process is Poisson, we can use the superposition principle to aggregate all the sources together. The interarrival rate will be eight times that of a single source. Based on the M/D/1 model, we can easily calculate the average queue size or the buffer size of the hub, and the average delay time of a packet from the equations below [49].

The average number of packets, N, in the system is,

$$N=\frac{\lambda}{\mu-\lambda}\left(1-\frac{\lambda}{2\mu}\right) \tag{3. 1}$$

Therefore, the average queue size, *L*, will be the average number of packets in the system minus the average number of packets being served, as follows,

$$L=N-\frac{\lambda}{\mu}. \tag{3. 2}$$

The average delay time for each packet from the source to the destination is,

$$T=\frac{N}{\lambda}. \tag{3. 3}$$

Therefore, the average waiting time at the queue is the average delay time minus the time spent in the processing,

$$W=T-\frac{1}{\mu}. \tag{3. 4}$$

From these equations, the waiting time at the queue increases exponentially, as indicated in Table 3.1 and Figure 3.3. Notice that the waiting time goes to infinity as the interarrival time of the packet approaches 1 second as in Figure 3.3. This is because the interarrival time is approaching the time required to process one packet.

Interarrival time (sec)	Arrival rate (1/s)	System Mean Pkt Val.	Queue Size	Average Delay (sec)	Average Delay (log(sec))
100	0.01	0.001	5e-5	1.005	0.002166
80	0.0125	0.001	0	1.01	0.004321
50	0.02	0.02	0	1.01	0.004321
20	0.05	0.05	0	1.03	0.012837
10	0.1	0.11	0	1.1	0.041393
8	0.125	0.13393	0.01	1.07143	0.029964
5	0.2	0.225	0.025	1.125	0.051153
2	0.5	0.75	0.25	1.5	0.176091
1.1	0.90909	5.45455	4.54545	6	0.778151
1.001	0.999	500.5	499.5	501	2.699838
1.0001	0.9999	50000.5	49999.5	50001	4.698979

Table 3.1: Analytical results based on the M/D/1 model.

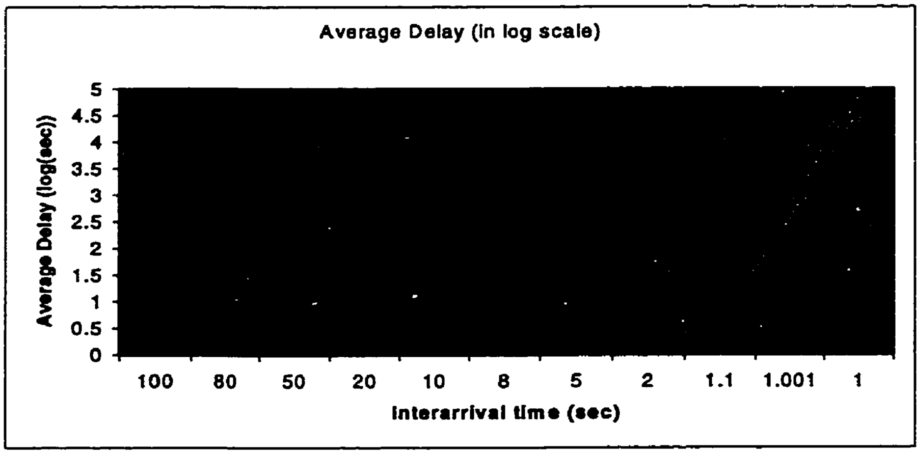


Figure 3.3: Average packet delay at the queue verses packet interarrival time.

3.3.2 The COMNET III Model

Each satellite node has a traffic source that generates 9600 bits at the specified interarrival time. The satellites are represented by eight nodes, Node0 to Node7, and they are connected to the center Hub through the point-to-point links. The network layout is illustrated in Figure 3.4.

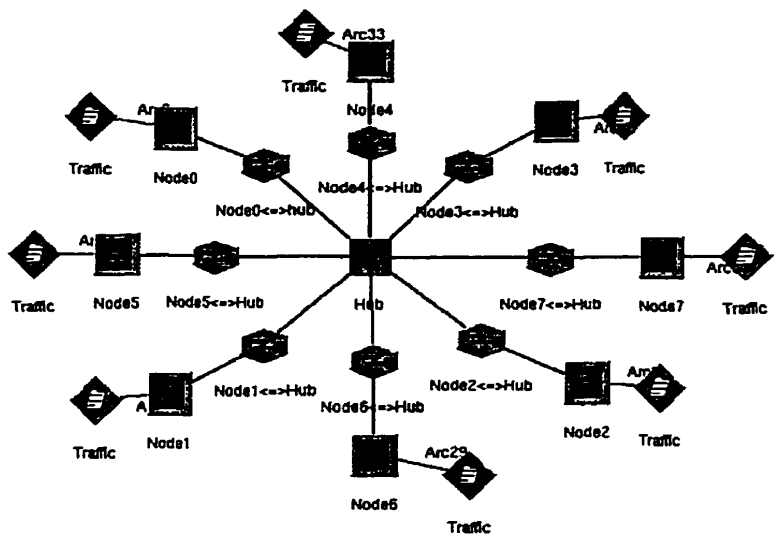


Figure 3.4: COMNET III satellite network.

The model was simulated 12 times with different interarrival times (in seconds). The results of the simulation are presented in two groups. The first group consists of the first nine columns of Table 3.2, the interarrival times are 100, 80, 50, 20, 10, 8, 5, 2, and 1. The second group consists of the last three columns in which the results are not valid, more explanations will be provided.

Interarrival time (sec)	100	80	50	20	10	8	5	2	1	0.5	0.2	0.1
Simulation time (sec)	10000	8000	5000	2000	1000	800	500	200	100	50	20	10
Packet generated	698	698	698	697	697	695	694	691	674	392	152	72
Packet received	698	698	698	697	697	695	694	691	674	392	152	72
Average packet received	0.07	0.087	0.14	0.349	0.697	0.869	1.388	3.455	6.74	7.84	7.6	7.2
Average delay (sec)	1.67	1.67	1.69	1.71	1.75	1.79	1.84	2.13	3.55	9.81	5.07	1.39

Table 3.2: COMNET III simulation results for the satellite network.

	Warm up time = 0 sec			
Arrival rate (/sec)	1	2	5	10
Sim. time (sec)	100	50	20	10
Pkt. generated	766	450	231	158
Pkt. received	766	450	231	158
Ave. pkt. received	76.6	9	11.55	15.8
Ave. delay (sec)	6.08	11.43	4.97	1.78

	Warm up time = 100 sec			
Arrival rate (/sec)	1	2	5	10
Sim. time (sec)	100	50	20	10
Pkt. generated	754	491	246	178
Pkt. received	754	491	246	178
Ave. pkt. received	75.4	9.82	12.3	17.8
Ave. delay (sec)	5.86	46.33	53.98	40.12

Table 3.3: Simulations with and without warm up.

The simulation time is scheduled in the way that each satellite node is expected to generate a hundred packets. In other words, the total number of packets generated in a simulation run should be 800 for the eight satellites.

Table 3.2 shows that COMNET III cannot perform the required task as the interarrival time is reduced to less than 1 second. The simulation results in Table 3.3 indicate that the problem is not caused by insufficient warm up time for the random generation process to reach steady state. The result for using a reasonable period of warm up time is very close to the one with no warm up time. Figure 3.5 shows that the number of packets generated drops exponentially without any known reason. The results of the simulation from the last three interarrival times are not relevant. The potential problem with this is that in a large simulation there is no indication when the results are totally incorrect.

The results also indicate that no packets are dropped. The number of packets received is the same as the number of packets generated. The average delay time versus the average number of packets generated per second is shown in Figure 3.6. The figure shows that the delay time increases exponentially as the number of packets generated in a second increases.

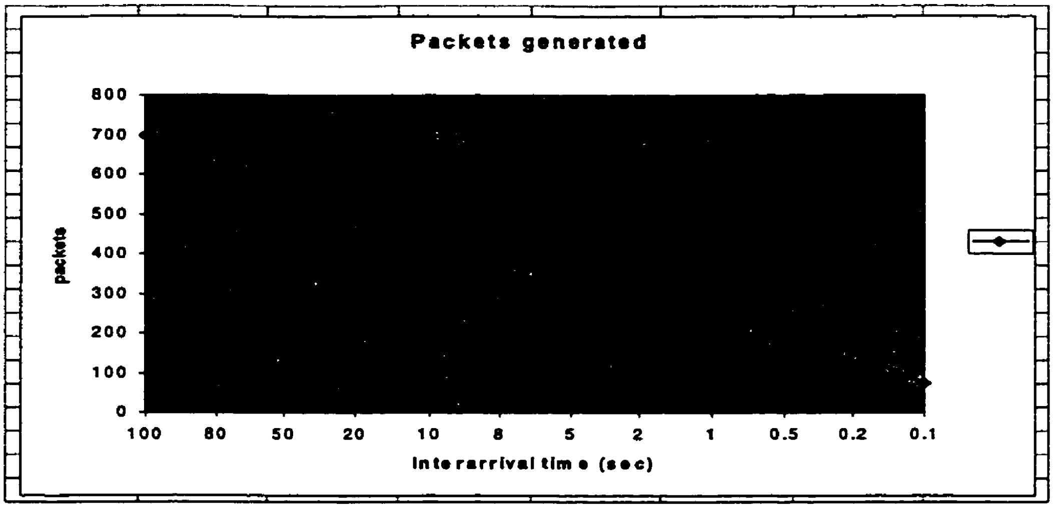


Figure 3.5: Packets generated during a simulation run.

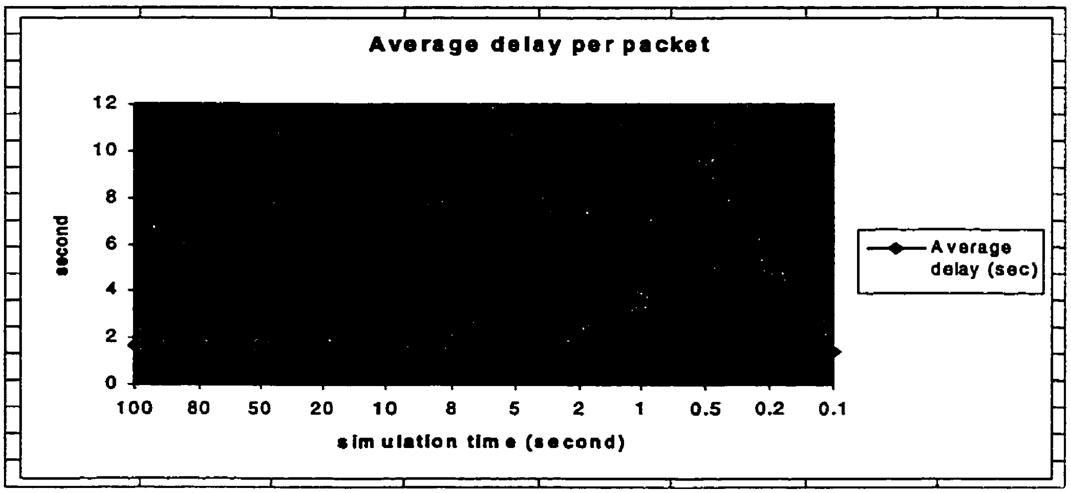


Figure 3.6: Average delay time per packet.

The dropping of the delay time for the last three simulations was caused by the fact that the traffic sources could no longer provide the required traffic load. The queue size reduces as the traffic load decreases. Even though the maximum load occurs at the interarrival time equal to 0.5 seconds, the simulation is not valid. The exponential increase of the delay time is based on extrapolating the results from the first group of the simulations. The second group of simulations does provide useful information. That is, when the traffic is too high, and the center hub has infinite buffer size, the delay time is increased linearly with the simulation time.

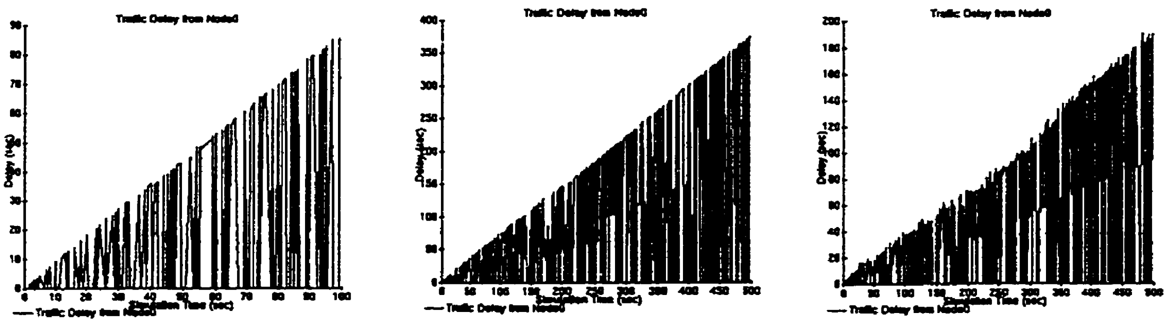
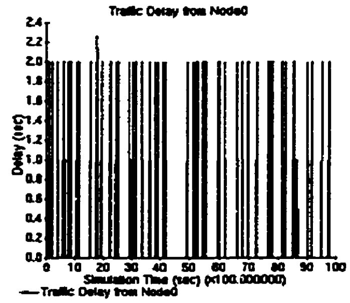


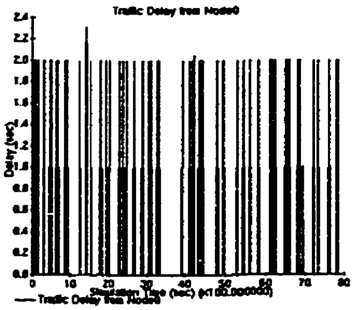
Figure 3.7: Simulation results of the second group.

The figure above shows the delay recorded from a satellite node for the second group of interarrival times, 0.1, 0.2, and 0.5 seconds respectively. The simulation time is adjusted so that the increment of the delay can be observed. The graphs clearly show that the delay time increases linearly with the simulation time. The longer the simulation time, the longer the delay becomes.

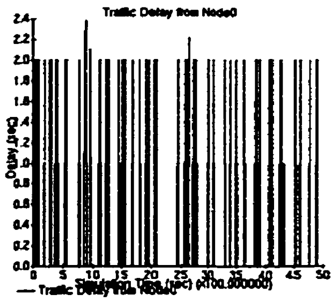
The packet delay recorded from the same satellite node for the first group of interarrival times is illustrated in Figure 3.8, for 100, 80, 50, 20, 10, 8, 5, 2, and 1 seconds.



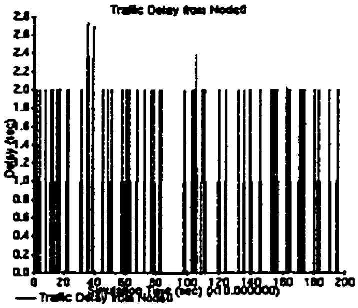
(a) 100 seconds



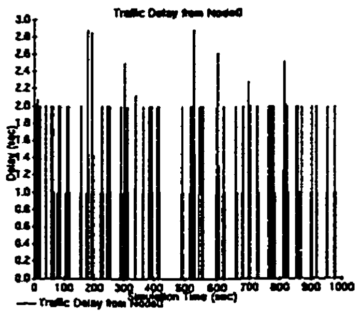
(b) 80 seconds



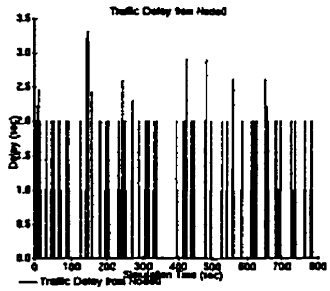
(c) 50 seconds



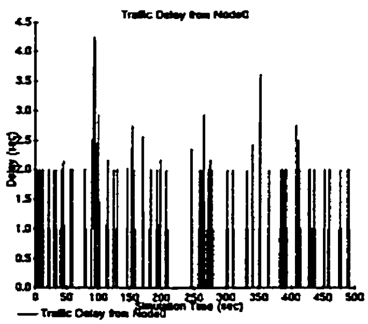
(d) 20 seconds



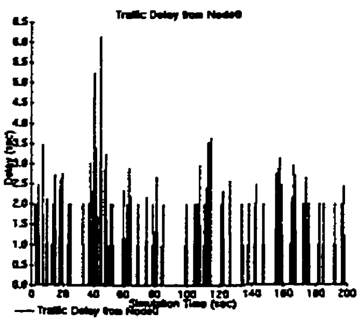
(e) 10 seconds



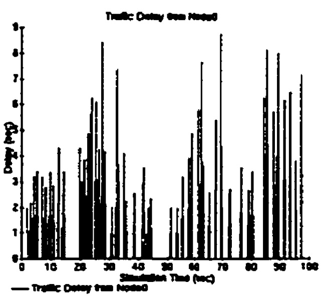
(f) 8 seconds



(g) 5 seconds



(h) 2 seconds



(i) 1 second

Figure 3.8: Simulation results for different interarrival times.

3.3.3 The OPNET Model

The satellites and the hub are represented by the nodes as shown in Figure 3.9. Each satellite is connected to the hub by a bi-directional point-to-point link representing a channel.

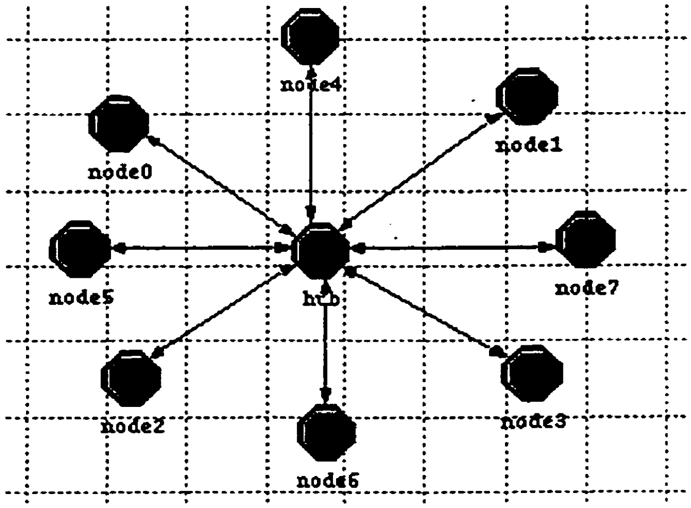


Figure 3.9: OPNET satellite network.

The hub node is used as a relay station only, so it is constructed with eight pairs of transmitters and receivers and with a processor performing routing of the packets.

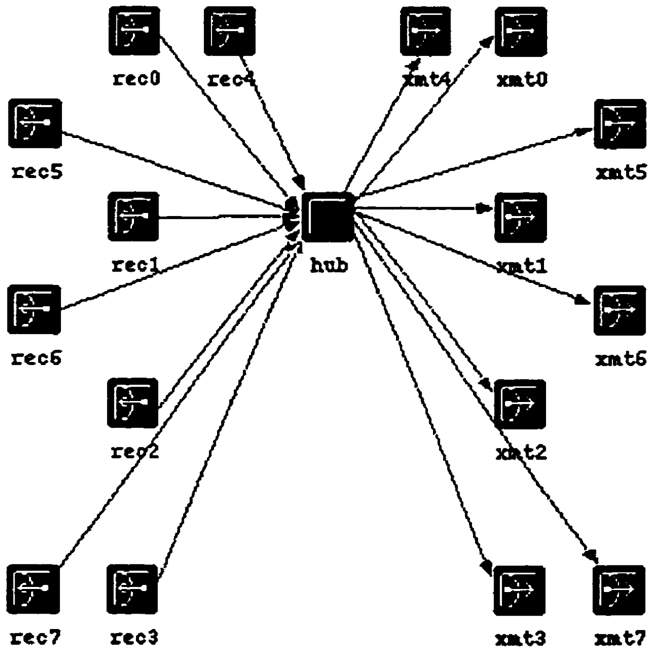


Figure 3.10: Hub node.

The satellite nodes also have a pair of transmitter and receiver at each node to connect to the hub. Every satellite node has an ideal packet generator, and a processor to handle transmitting and receiving of the packets.

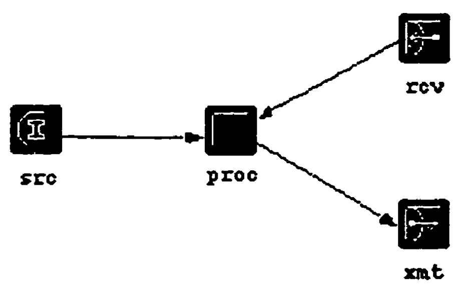


Figure 3.11: Satellite node.

The operation of the processors can be looked at as a state machine. At the beginning of the simulation, the nodes are at the init state and go to idle during simulation. When a packet arrives at the satellite node from the ideal generator, the processor goes to xmt state. It assigns the destination address to the packet, and increments by one the global variable tx_pkts, which keeps track of the number of packets being sent to the hub, then it goes back to idle. When the node receives a packet, the processor goes to the rcv state. It calculates the time delay between transmitting and receiving of the packet. The processor also adds the delay time to the global variable total delay and increments the global variable rcvd_pkts which keeps track of the number of packets received in the network, then goes back to idle.

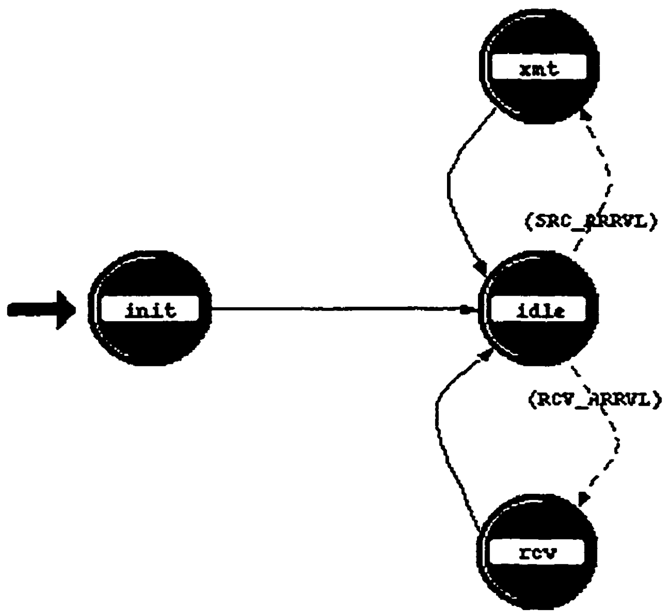


Figure 3.12: The satellite node as a finite state machine.

At the hub node, when the simulation begins, the hub processor is in init state where all the global variables are initialized. The variables, rvcd_pkts, tx_pkts and total_delay, are set to zero. During the simulation the hub processor stays idle, unless interrupted by the incoming packets or by the END_SIM signal. When a packet arrives, the hub will go to route_pk state, which gets the packet and reads the address of the destination, then sends the packet to the right path and goes back to idle. When an END_SIM signal is received, the simulation terminates. The hub processor then calculates the average transmitted packets and average received packets over the simulation period, and the average delay time per packet. The results are stored in a file.

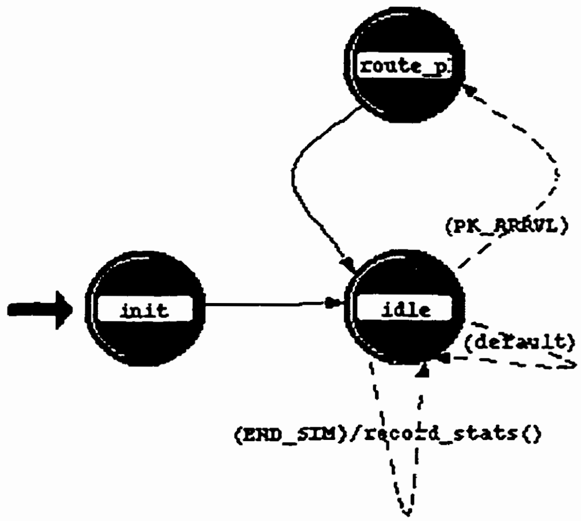


Figure 3.13: Hub node finite state machine.

The results of a new simulation run will append to the same file. The model was simulated 12 times with 12 different interarrival times (in seconds) the same as those used in the COMNET III model. The obtained results are displayed in Table 3.4.

Interarrival time (sec)	100	80	50	20	10	8	5	2	1	0.5	0.2	0.1
Simulation time (sec)	10000	8000	5000	2000	1000	800	500	200	100	50	20	10
Packets generated	830	811	830	830	830	830	830	830	830	830	830	830
Ave. pkt generated (/sec)	0.083	0.101	0.166	0.415	0.83	1.038	1.66	4.15	8.3	16.6	41.5	83
Ave. pkt received (/sec)	0.083	0.104	0.166	0.415	0.827	1.038	1.65	4.11	6.96	7.02	6.2	5.1
Average delay (sec)	2.008	2.01	2.016	2.054	2.109	2.135	2.234	3.085	9.706	15.86	9.44	5.3

Table 3.4: OPNET simulation results for the satellite network.

The simulation results show that as the interarrival time becomes smaller, which means the traffic becomes heavier, the throughput of the network decreases. The number of packets received decreases. The problem becomes severe when the interarrival time is reduced to 0.1 second. The throughput is less than 10% of the generated packets as in the following figures. The delay of a packet from the sender satellite to the receiver satellite is also increased as the traffic becomes heavier.

The average delay per packet in the figure above shows that the delay has an exponential increment. Compare Figure 3.14 with the delay record from the COMNET III model in Figure 3.6, both models give similar results with a condition that the simulation result from the second group of interarrival times is not considered. The time delay of the packets for 12 different interarrival times are included in Figure 3.16.

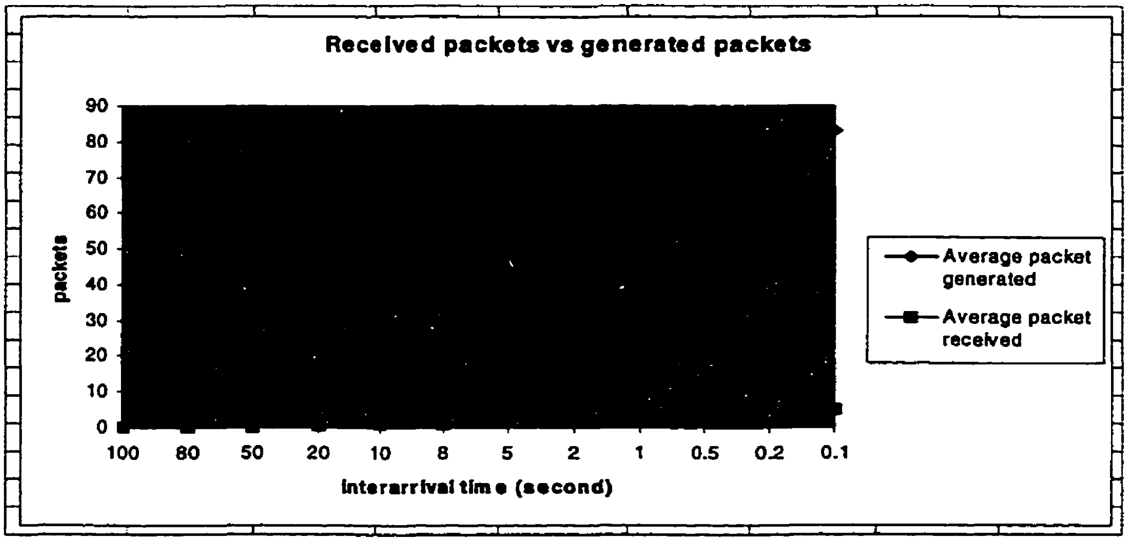


Figure 3.14: The throughput of the satellite network.

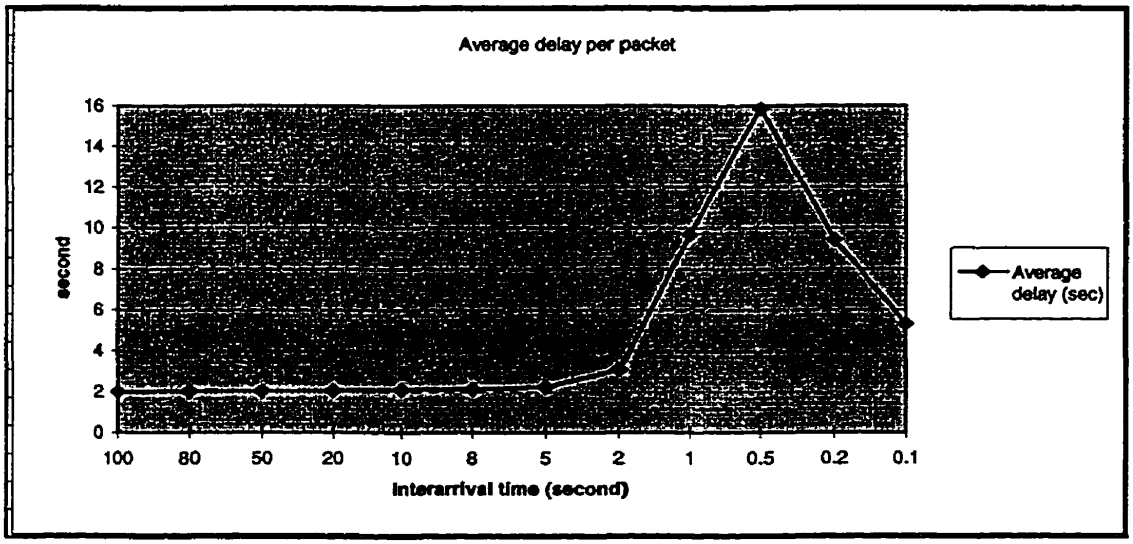


Figure 3.15: The average delay per packet for each simulation run.

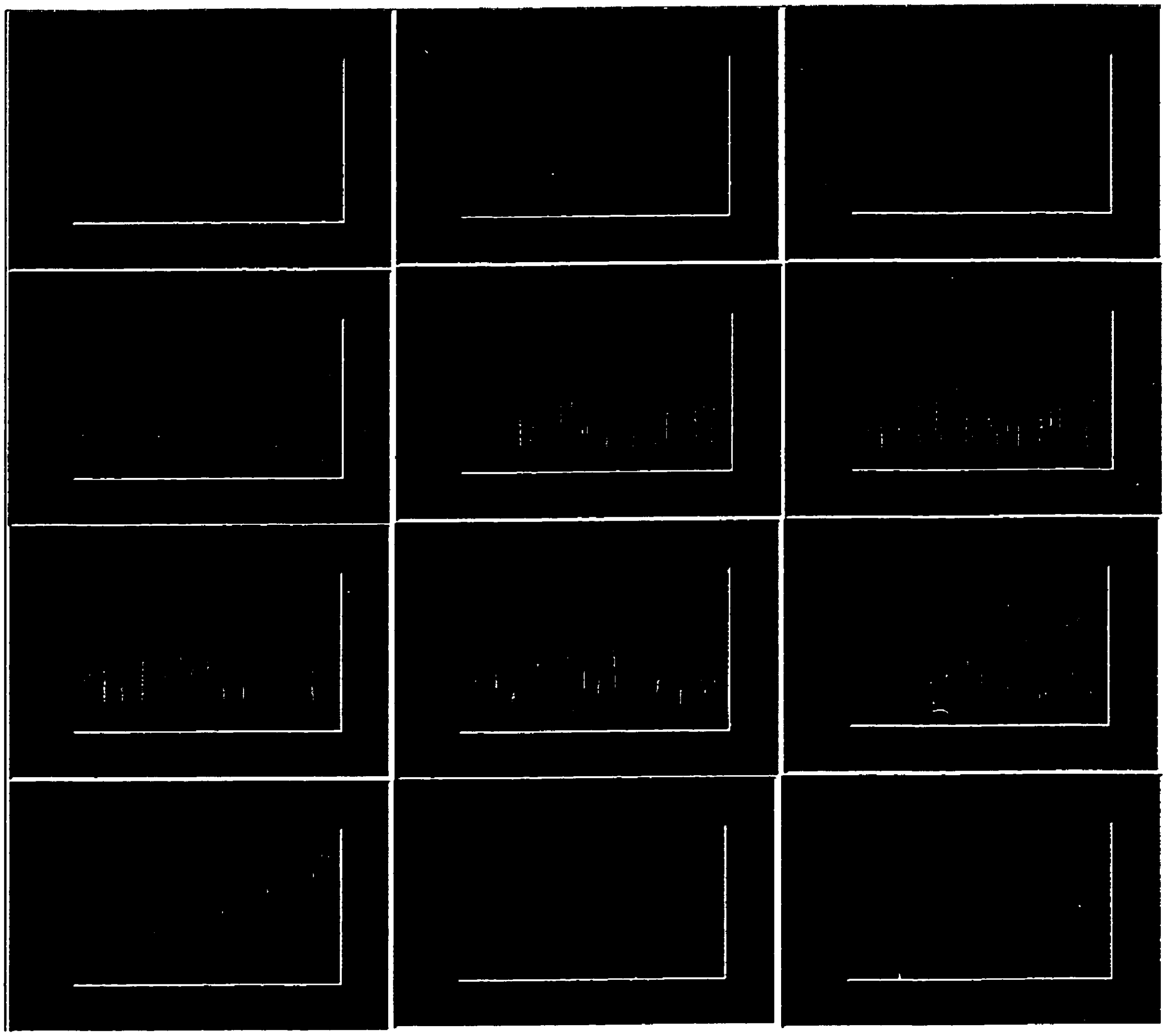


Figure 3.16: The time delay for each simulation.

3.3.4 Conclusion and Summary

A satellite model has been built using both the network modeling and simulation tools. Both satellite models were built in such a way that they are approximately equal to each other for evaluation purposes. The analytical results were derived to validate the simulation results from the tools.

The analytical model indicated that there is a boundary on the traffic load which must not exceed the processing capacity, but the simulation results do not agree with this. The reason for this is that the OPNET model eventually drops packets as indicated in Figure 3.14, where the average number of packets received is smaller than the average number of packets generated.

The recorded statistics from the simulations indicated that the COMNET III modeling software is affected by the interarrival rate of the packets. The simulation results are not valid for high interarrival rates. The software is not able to generate the required number of packets as would be expected by the user while running a simulation. The results indicate that although simulation is a powerful engineering tool, analytical checks and knowledge of reasonableness checks are required by the user as opposed to solely relying on the results of a simulator.

Chapter 4 Link Layer Simulation

This chapter demonstrates the modeling methods on the OSI link layer using the two simulation tools. This chapter provides an overview of communication networks and types of link layers currently available.

4.1 Types of Communication Networks

A communication or computer network consists of a number of geographically distributed users connected by a communications facility and the connections are arranged according to a network topology. The term user denotes an access point to the communication facility, and it could be user-terminals, or computers with which the terminals interact.

Many types of networks are in place presently. The types of the networks are usually defined by the network size [19]. In general, networks are classified into LAN, Campus Area Network (CAN), MAN, WAN, and Global Area Network (GAN), as described in Table 4.1.

Network	Distance	Classification
Local Area Network (LAN)	0.1 km	Building
Campus Area Network (CAN)	1 km	Same area buildings
Metropolitan Area Network (MAN)	10 km	Regional
Wide Area Network (WAN)	100 – 1000 km	National
Global Area Network (GAN)	1000 km or more	International

Table 4.1: Types of Networks.

The public network is a mixture of different communication technologies. It is very common to have a public network consisting of Frame Relay, Switched Multi-megabit Data Service (SMDS), ATM, and BISDN. Detailed descriptions of these technologies are contained in the coming sections. Similarly, it is common that a LAN is built on top of a Distributed Queue Dual Bus (DQDB), Fiber Distributed Data Interface (FDDI), ATM and Synchronous Optical Network (SONET) technologies.

4.2 Types of Links in Local Area Networks

The most popular types of links in LANs are ALOHA, Carrier Sense Multiple Access (CSMA), Carrier Sense Multiple Access with Collision Detection (CSMA/CD), Point-to-Point, and Token Passing. This section describe the functions of ALOHA, CSMA/CD, Token Bus of the token passing and Point-to-Point. Modeling and simulation of these types of LANs will be covered in the next section.

4.2.1 Point-to-Point

A point-to-point link connects two nodes together such as satellite links or ad hoc connections. In data communications, the point-to-point link is a full duplex bi-directional communications path characterized by the speed and number of channels. In circuit-switched traffic and satellite communications, separate bandwidth and/or channels characterize the point-to-point links. In the next section, a geo-stationary satellite network is used as a demonstration.

4.2.2 ALOHA

An ALOHA link is a multi-access link to which several nodes may be connected. It is a random access radio link where random transmissions may occur. The transmission occurs whenever traffic is generated. Collisions are detected after the transmission with rescheduled retransmission. Transmissions are sent without knowing whether the link is idle or not. In Aloha, if the sender gets acknowledgement from the device it is trying to reach, it continues to transmit. Otherwise (as if in the case of collision), the sender will start over again.

4.2.3 CSMA/CD

A CSMA/CD link is a CSMA with a collision detection feature called the Collision Detection (CD part of the CSMA/CD). CSMA can detect if a link is busy and delay its transmission until the link is idle. There is no collision detection implemented in CSMA, and it relies on time-outs to indicate that a frame was not delivered and schedule retransmission after a random delay. In a CSMA/CD LAN, if two or more nodes begin transmitting at the same time and cause a collision, they will both back off for a different amount of time (based on the Backoff algorithm) before attempting to retransmit. The well-known Ethernet and IEEE 802.3 standards both use CSMA/CD.

4.2.4 Token Ring

A token ring LAN can be wired as a circle or a star. The most common wiring scheme is called a star-wired ring. In this configuration, each computer is wired directly to a device called a Multiple Access Unit (MAU). The MAU is wired in such a way as to create a ring between the computers. If one of the computers is turned off or its cable is broken, the MAU automatically recreates the ring without that computer.

A token ring LAN always works logically as a circle, with the token passing around the circle from one node to the another. Token ring LANs can operate at transmission rates of either 4 megabits per second or 16 megabits per second. The number of computers that can be connected to a single LAN is 256.

4.2.5 Ethernet

Ethernet is a bus-based local area network technology that is in common use today. The network access process is managed by a Media Access Control (MAC) protocol standardized by the IEEE under the name 802.3. The role of this MAC protocol is to provide efficient and fair sharing of the communication channel which is the bus connecting the stations of the LAN.

The Ethernet MAC accepts data packets from a higher layer protocol, the network layer, and attempts to transmit them at appropriate times to other stations on the bus. Because the higher layer protocols can forward data at any time and the bus is a broadcast medium, it is possible that several stations will simultaneously attempt to transmit, thereby interfering with each other. This event is referred to as a collision and results in the loss of all packets involved. The Ethernet MAC protocol uses carrier sensing and deference mechanisms to handle collisions.

Carrier sensing is a mechanism used to prevent collisions when a station knows that the bus is already in use. Listening to the bus for activity and withholding transmissions until after activity has ceased does this. This mechanism is called deference and is limited in its effectiveness by the propagation delay of information signals on the bus. In the worst case, once a transmission has begun, the time required for a signal to travel the full length of the bus must elapse before all stations can be "warned" that the bus is busy. Until that time, a station at the opposite end of the bus would falsely assume that the bus is available and may initiate a transmission that would result in a collision.

4.3 The ALOHA Network

This subsection compares the objects in the COMNET III object library with analytical results, and alternatively using the OPNET object library. The goal is to validate the credibility of these objects.

In an ALOHA network, multiple users may share the same channel. The users send bursts of data, called packets, whenever they wish. If two or more bursts overlap on time (i.e. a collision), the users involved retransmit their packets after a random time delay. It is hoped that a second collision will not occur. If it does occur, the retransmission is repeated until each party is successful. This technique has the advantage of being relatively inexpensive to implement. In slotted Aloha the time is segmented into time slots, and transmission is only allowed at the beginning of a slot [49]. This section will compare the performances of the pure aloha 1-persistent networks using different tools. The OPNET ALOHA network consists of one receiving station and ten transmitting stations. The performance can be determined by the throughput measured at the receiving station.

4.3.1 The Theoretical Model

Let us consider the transmission of a packet in a pure ALOHA system. If the packet transmission period is P seconds, as shown in Figure 4.1, the packet is vulnerable during a period of $2P$ seconds in the sense that any other packet transmission generated and initiated in the vulnerable period will transmit, collide with, and destroy our transmitted packet. Let S denote the throughput of the satellite channel (average number of successful transmissions per transmission period P), and let G denote the average channel traffic (measured in the number of packet transmissions attempted per P seconds).

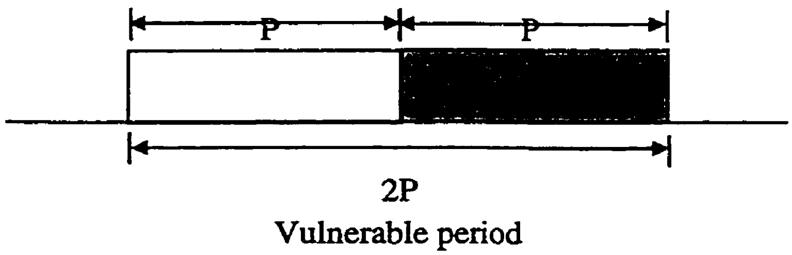


Figure 4.1: The vulnerable period for pure ALOHA.

If we assume that the total traffic G entering the channel is an independent process generated by an infinite population of users generating at an infinitesimally small rate, then we have

$$S = GP, \tag{4. 1}$$

with the assumption that no additional packets are generated during the vulnerable period. If we assume that the channel traffic is Poisson, then we have $P=e^{-2G}$, which gives

$$S = Ge^{-2G} \tag{4. 2}$$

It is easily seen that the maximum throughput one can obtain from a pure ALOHA channel occurs at a value of G=1/2 and yields a maximum efficiency of 0.184, see Figure 4.2.

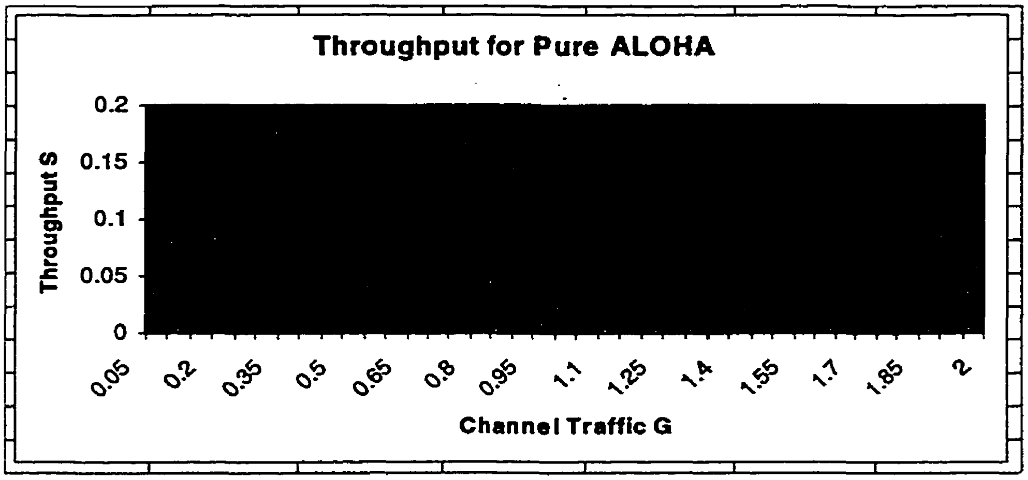


Figure 4.2: Throughput for pure ALOHA.

4.3.2 COMNET III Models

An ALOHA link from the object library is used to connect all the stations, f_1 to f_10, using the same channel. The bandwidth of the channel is 1 kbps. Each message source is generating fixed sized packets following an exponential distributed interarrival time. The size of the messages is 1 kilobit. When a station receives a packet, it will transfer the packet to the receiving station f_0 through the ALOHA link.

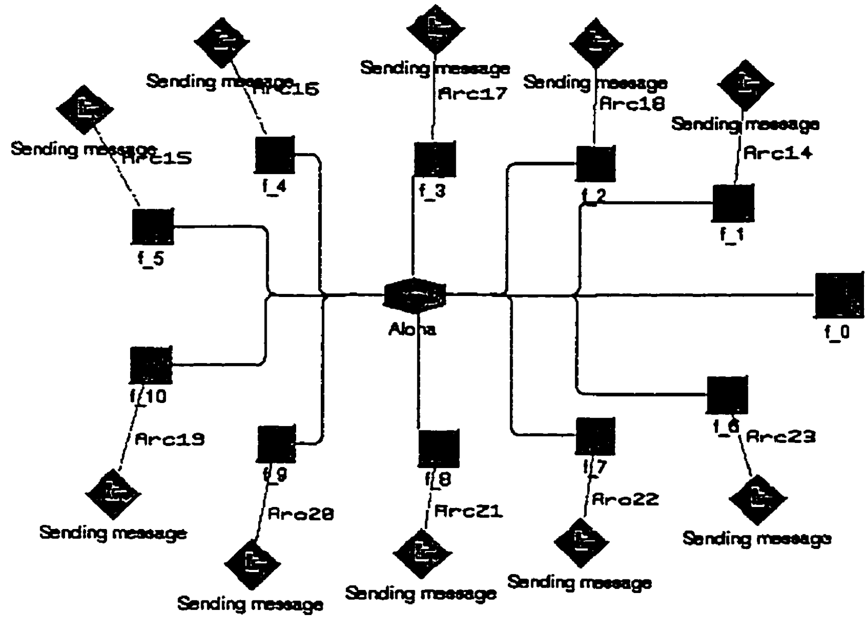


Figure 4.3: 1-persistent Aloha network.

The simulated results from the COMNET model are based on the IEEE exponential backoff algorithm with different traffic load. The COMNET III tool generated a very complicated report from every simulation run. The report contained all the information regarding an ALOHA link. Alternatively, the user can use the total average link efficiency (%) to determine the throughput. This is valid only when the link efficiency also represents the packet throughput. For example, a 100% efficiency means the link utilization is being used at 1kbps or 1 message per second on average, thus S=1. The traffic load is determined by the average number of messages assembled (in one second).

Sim time (sec)	Msg Assembled	Collision Episodes	Collided Frames	Traffic load (G)	Average delay (sec)	Throughput (S)
20000	398	10	121	0.0199	433	0.0194
4000	397	53	841	0.0993	471	0.0860
3000	415	49	815	0.1383	483	0.1220
2000	404	139	2238	0.2020	1134	0.1325
1800	402	129	2340	0.2233	1225	0.1517
1200	410	225	5624	0.3417	2437	0.1542
1000	448	301	12017	0.4480	7037	0.1470
800	372	253	7825	0.4650	4192	0.1488
600	408	390	14584	0.6800	10523	0.0300
400	379	379	17086	0.9475	9882.5	0.0000

Table 4.2 : COMNET III ALOHA simulation results.

The figure below shows the link efficiency versus the traffic load from Table 4.2.

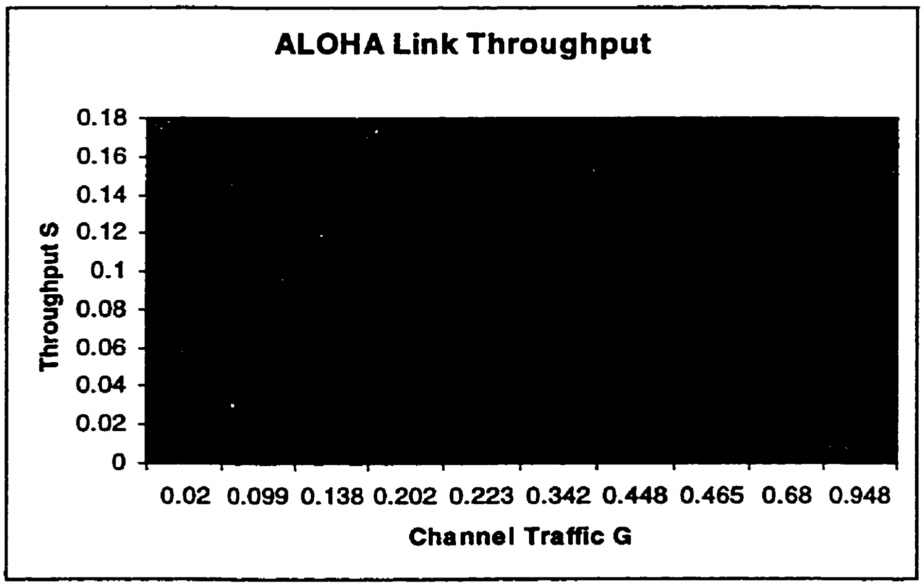


Figure 4.4: The link efficiency of the COMNET III ALOHA model.

4.3.3 OPNET Models

The ALOHA model has ten transmitter nodes, f_1 to f_10 as in Figure 4.5, to represent the ten users. The fact that it allows only one transmission at a time simplifies the network by using a receiver node to represent any user who is the receiver at the time. A receiver node, f_0, is used to symbolize the receiver for all the users. The duty of the receiver node is to monitor the transmission to ensure reliable retransmissions. When the receiver node receives a collision-free transmission, it acknowledges the sender through a separate channel. If the sender did not receive the acknowledgement after the time-out period, then collision is assumed and retransmission is performed. The sender will repeat the transmission after some random delay to avoid the same collision with other users.

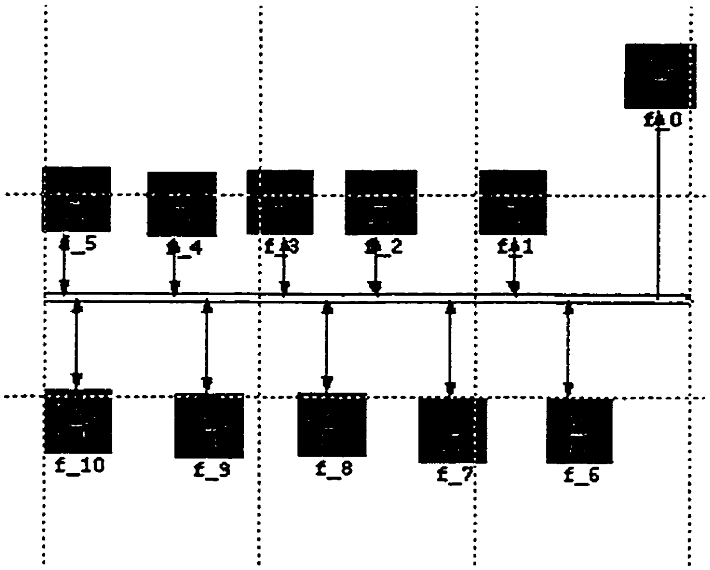


Figure 4.5: OPNET ALOHA network.

The user nodes are basically transmitting packets at the assigned rate. However, it is enhanced with a receiver bus module and a sink processor module as in Figure 4.6. The enhancement of the bus receiver is to support the eventual full duplex capability of the CSMA protocol for latter use in the CSMA network. The purpose of sink processor is to inform the network editor that the node may have full duplex capability. A statistical wire is used to connect the bus receiver to the processor module. The function of the statistical wire is to convey simple numeric signals or control information between modules. In this case, the statistical wire will inform the processor module of the busy status of the channel and also provide interrupts to the processor when the channel condition changes. For now, the statistical wire is disabled and is not used in the ALOHA model.

Basically, the ALOHA transmitter node is constructed by an ideal generator to produce traffic, a processor module, and a bus transmitter connects the users to the channel. The finite state diagram of the processor module is illustrated in Figure 4.7. The state diagram implies that the processor module is usually at idle state, but is interrupted by PKT_ARVL. When a packet arrives from the ideal generator, the processor module is forced to tx_pkt state and transmits the packet.

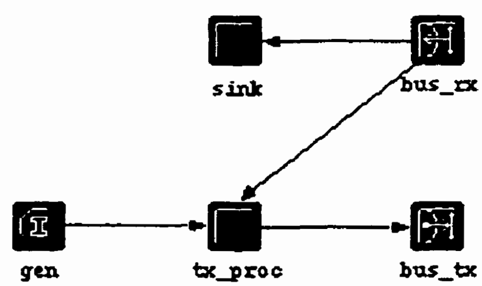


Figure 4.6: OPNET user node (enhanced transmitter node).

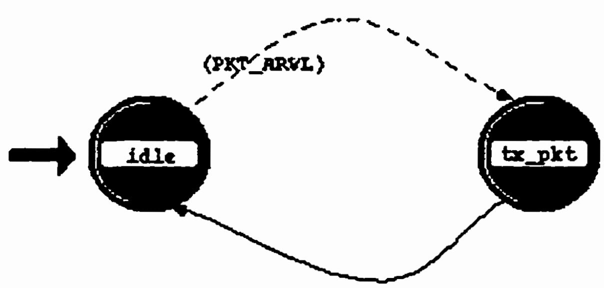


Figure 4.7: Finite state diagram of the processor module.

The received node is constructed by a bus receiver and a receiver processor module as in Figure 4.8. The bus receiver transfers all the received packets to the receiver processor module. The receiver processor is initially at init state. When the packet is received, the receiver is forced to idle state and performs a function called proc_pkt(). The function is to get the incoming packets, destroy the packets and count the number of received packets for statistical purposes. The transition from idle state to idle state when receiving a packet is carried out by the pro_pkt() function. The receiver processor module can also be interrupted by the END_SIM signal at the end of a simulation run. At the end of simulation the record_stat() function is performed. The function calculates the average number of transmitted packets from all the transmitter nodes and the average number of received packets at the receiver node, and writes the result to a scalar file. A set of new data will be generated for each simulation run. The data will then be appended to the same scalar file.

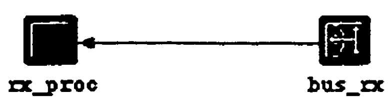


Figure 4.8: OPNET receiver node.

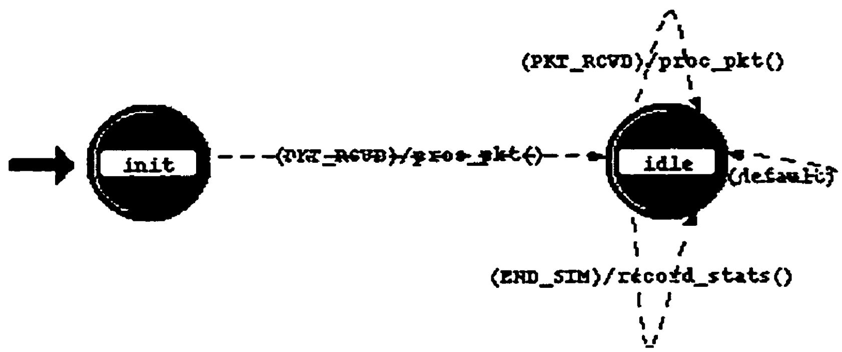


Figure 4.9: Finite state diagram of the receiver processor module.

The scalar file is plotted in Figure 4.10. The throughput is the average number of the received packets per second and the channel traffic is the average number of packets sent by all the users or transmitters on the network.

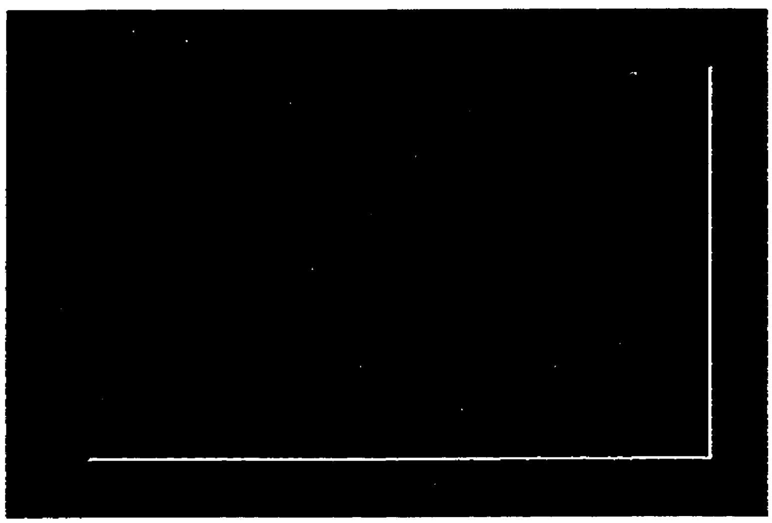


Figure 4.10: ALOHA throughput.

4.4 Summary and Conclusion

The chapter first describes different kinds of link layer medium access control mechanisms such as ALOHA, CSMA/CD, token ring, and Ethernet. The analytical results and the simulated results indicate that the ALOHA link object in the COMNET III library does not agree with the analytical result. The statistical wire in the OPNET library can approximate the ALOHA network with much more accuracy as shown in the figure below.

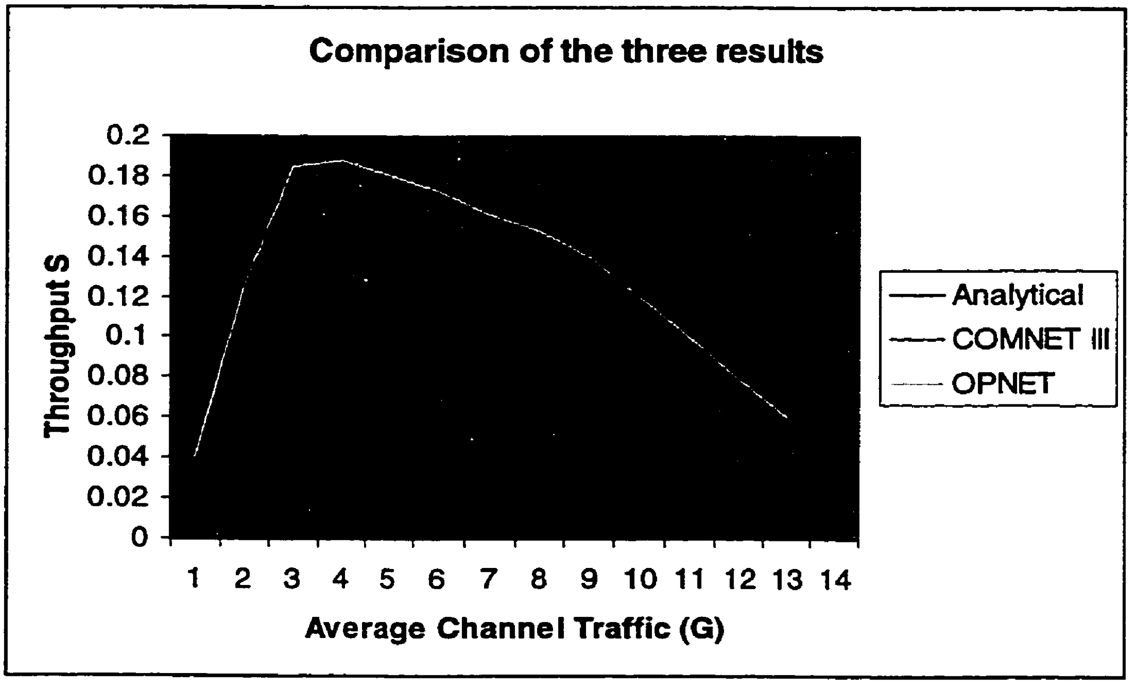


Figure 4.11: Comparison of the three results.

Chapter 5 Common Channel Signaling System 7 Simulation

The Common Channel Signaling System 7 (CCS7) is a signaling system that constructs the Intelligent Network (IN). The IN provides control and administrative functions for many Public Switching Telephone Networks (PSTN) and Integrated Service Digital Networks (ISDN). The CCS7 system for telephone networks is an individual packet switching network built on top of the circuit switching networks. The signaling links in the signaling network provide transport for the signaling messages, while the trunk groups provide transport for the calling traffic.

This chapter will model the CCS7 network that provides call setup for telephone calls and ISDN based services. In order to have a better understanding of the CCS7 model, this chapter starts with the description on ISDN and IN to help readers to have a better understanding of the model.

5.1 Introduction to ISDN

Integrated Services Digital Network (ISDN), a foundation of broadband communications [8] [9] [25], is not a service but a technology to move information across the public telephone network by making all transmission circuits end-to-end digital with out-of-band signaling. ISDN provides an internationally accepted standard for transferring voice, data, and signaling. ISDN technology consists of two kinds of connections, namely, Basic Rate Interface (BRI) and Primary Rate Interface (PRI) [17].

5.1.1 Out-of-band Signaling

ISDN utilizes the out-of-band signaling to carry the signaling information. In other words, the signaling system has its own link, and is separated from the data link. If a data link has failed, the signaling link can still inform the necessary parties about the network failure. Signaling information refers to the information to setup and disconnect a call. For example, the time of the day a call is made, the number called, and the format of the data. This information is carried through a separate link to set-up and to control the calling.

5.1.2 BRI and PRI

In telephone networks, each pair of wires contains three channels, two B channels for voice and data, and a D channel for signaling. This kind of ISDN connection is called Basic Rate Interface (BRI). Another ISDN connection is called the Primary Rate Interface (PRI) which consists of 23 B channels and a D channel using a T-1 link.

BRI equips a single telephone circuit for multiple use. Customers can send fax and talk at the same time on the same line. However, a BRI line costs a little more than a regular business line, and extra costs on the equipment make BRI too expensive for residential use. Furthermore, BRI premises equipment is not widely available and usually it is hard for a customer to setup and use.

PRI dedicates one of the 24 channels for T-1 signaling, and the other 23 channels to transmit data and voice. Many of the large businesses have PRI services provided by long distance carriers. A PRI is easier to implement than a BRI. Long distance carriers would typically provide ISDN PRI service only to the directly connected T-1 customers. The single line customers are the candidates for BRI, but usually the long distance carrier has no direct connection to these customers. The single line customers are connected to the call-switches of the local telephone company. The following figure shows how a PRI is employed in the telephone network.

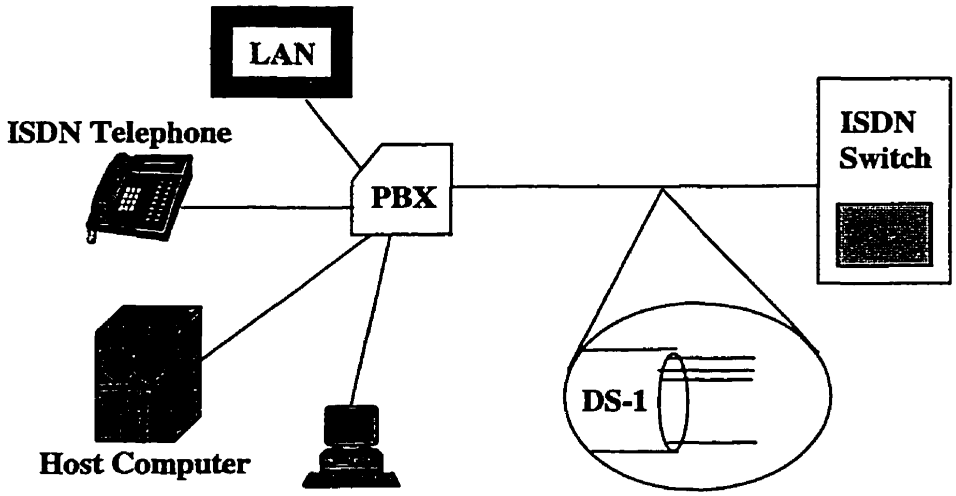


Figure 5.1: ISDN Primary Rate Interface (PRI).

5.1.3 Imperfections of ISDN

The growing demands of bandwidth for links among computers, video terminals and imaging systems calls for frame relay, ATM and other high-speed transport modes that will outpace narrowband ISDN. In 1993, there were 300,000 basic-rate ISDN links installed in the United States, and the growth rate is expected at 50% per year over the next few years [11]. The reasons for the growth are the increased bandwidth at the desktop, widespread popularity of graphics-based software, and multimedia applications. These have dramatically increased the amount of data users generate and need to exchange. Many users are also spoiled by the speed of their office LAN and find even the 28.8kbps modems tedious and slow. Although ISDN can provide better service compared to the basic telephone line and allow customers to access remote LANs or the Internet from home (telecommuting), the bandwidth is still limited. Its broadband counterpart, Broadband ISDN can provide a better service.

5.2 Intelligent Networks

An Intelligent Network (IN) element controls the heart of a network. IN elements can offer telecommunication operators a rich set of means to manage their network services and create new value to their customers. Traditionally, services were introduced by modifying software in switching systems. IN allows service logic resources and data to be distributed over network elements such as Service Control Points, Intelligent Peripherals, Service Switching Points and Signal Transfer Points. The architecture of the IN is illustrated in Figure 5.2 and followed by the outline of the functions of each element.

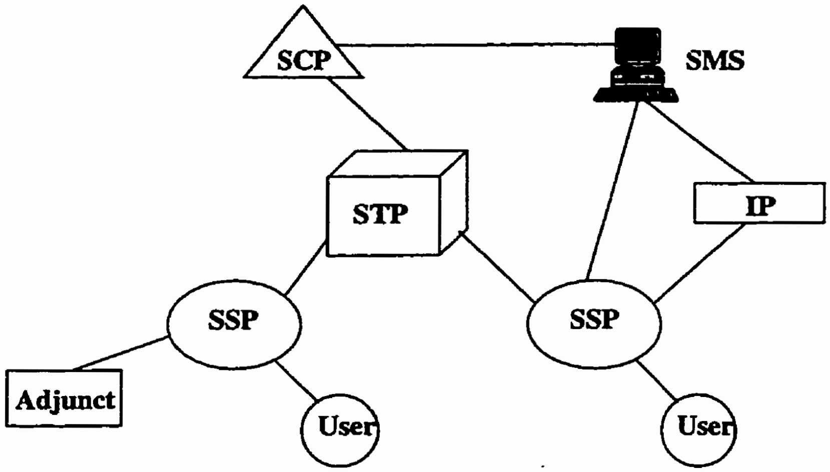


Figure 5.2: Intelligent Network (IN) architecture.

The different components are the following.

- SMS - Service Management System (usually owned by the network operators). This system updates the SCPs with data or programs and collect statistics from them. It enables the terminal linked to the SMS. The SMS uses commercial software, and is usually put on a mainframe, that provides a development environment for new services.

- SCP - Service Control Point. This is used when new IN services are introduced into the network and activated. If a service is based on functional components, the functional components are elected using a service logic interpreter. It should be able to access databases efficiently and reliably, and provide a software platform for rapid service creation.

- STP - Signal Transfer Point. This is part of the Signaling System No. 7 (SS7) Network. SS7 is a standardized communication interface through which the goal of the multi-vendor SCP and Service Switching Point (SSP) can be achieved. STP switches

SS7 messages to different SS7 nodes. The use of standalone or integrated STP depends on specific network configurations.

- SSP - Service Switching Point. This is an access point for service users and executes heavily used services. The higher layer protocol of SS7, namely the Transaction Capabilities Application (TCAP), is used to communicate between the SSPs and the SCPs.
- IP - Intelligent Peripheral. This system provides enhanced services. The functions of an IP are either not found in an SSP or too expensive to put in all SSPs. For example, speech synthesizing, voice messaging, speech recognition, or user data base information. By making IP physically separate from SSP but connected via a standard interface enables any vendor to own and operate an IP thus enabling the service providers to provide value-added services to its users.

5.3 The Common Channel Signaling System No.7

The Common Channel Signaling System No. 7 can be defined as the system that controls exchange points, network databases, and other nodes of a network that relate to the call setup, supervision, and call disconnect processes. The first introduction of Common Channel Signaling was based on the Consultative Committee on International Telegraph and Telephone (CCITT) Signaling System No. 6 (SS6) Recommendations. The SS6 protocol structure was not layered, but rather a monolithic structure designed to efficiently utilize the limited bandwidth of 2.4 kb/s. Signaling System No. 7 (SS7) was first introduced by CCITT in 1980 to provide a signaling system for digital trunks.

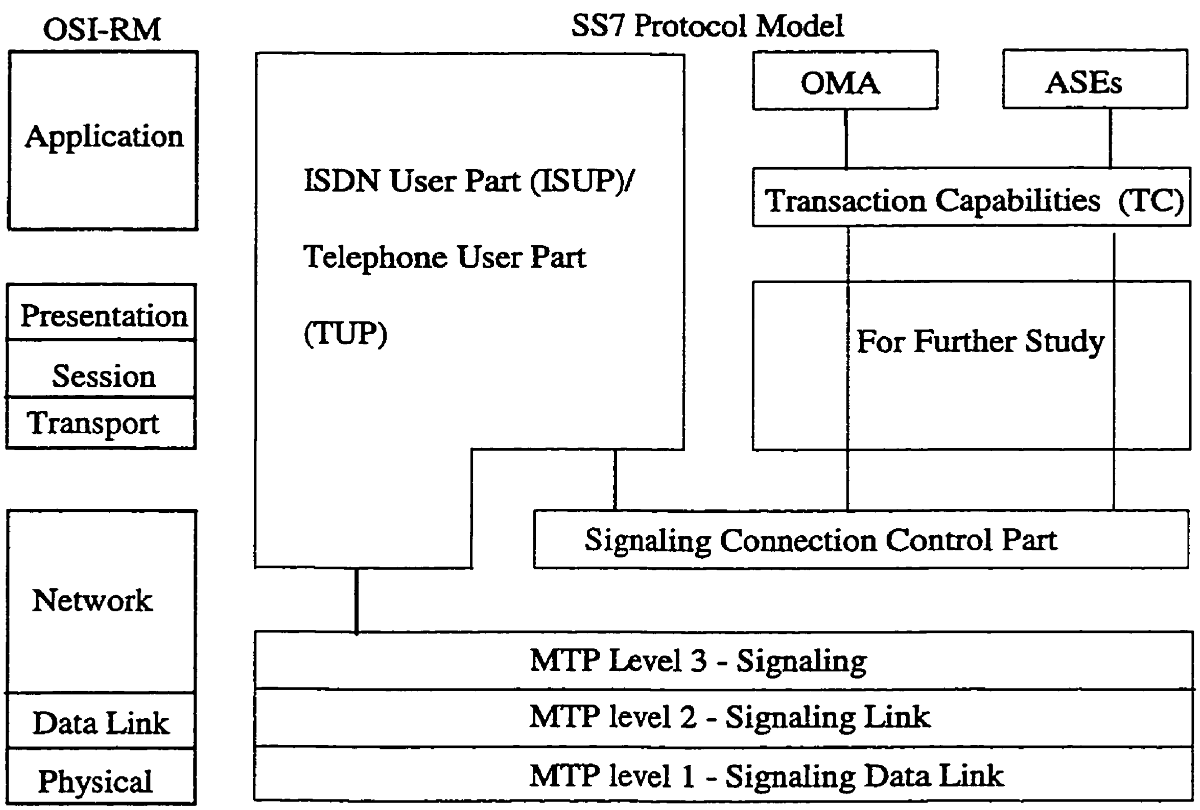
The objective of the SS7 is to provide an internationally standardized, general-purpose common channel signaling system. The Common Channel Signaling Number 7 (CCS7) is the signaling system in today's digital telephone network. The signaling system optimizes the operation of the telecommunication network via a controlled exchange. The coordination subsystem makes routing of the control signals through the network to perform call setup, call maintenance, call termination, and network management functions. The signals are arranged into small blocks of packets. The signaling system is implemented using packet-switching technology, although the communication network itself is a circuit-switched network.

The signaling network is overlaid on the circuit-switched network. The CCS7 performs three basic functions; supervising, alerting, and addressing. The supervising function monitors the status of a line or circuit to see if it is busy, idle, or requesting service. The voltage levels on signaling terminations or the on-off hook status of signaling tones or bits indicate the supervisory signals. An alerting function indicates the arrival of an incoming call. Alerting signals are bells, buzzers, woofers, tones, strobes and lights. Addressing is the methodology of transmitting, routing and destinating any signals over the network. Addressing signals are in the form of dial pulses, tone pulses or data pulses over loops, trunks and signaling networks.

5.4 SS7 Protocol Architecture

Instead of having seven layers as the OSIRM, SS7 is divided into four levels [26]. The first three levels form the Message Transfer Part (MTP) [44], which provides a reliable service to route the signaling messages between two Signaling Points (SPs) [48]. The

MTP is almost equivalent to the lowest three layers of the OSI reference model. The combination of MTP and the Signaling Connection Control Part (SCCP) at the fourth level is defined as the Network Service Part (NSP). The following paragraphs describe the functions of different levels. Figure 5.3 illustrates the architecture of the SS7 protocol and how the structure relates to the OSI layers.



OMAP = Operations Maintenance and Administration Part
ASE = Application Service Element

Figure 5.3: SS7 protocol architecture.

5.4.1 Level 1 – Signaling Data Link

A signaling data link is a full-duplex physical link interconnecting two signaling nodes. It corresponds to the physical layer of the OSI model. This level defines the physical, electrical, and functional characteristics of a signaling data link and the means to access.

5.4.2 Level 2 – Signaling link

It corresponds to layer 2 of the OSI model. The level is defined as a data link control protocol that provides for the reliable sequenced delivery of data across a signaling data link, and performs detection and recovery from transmission errors. It also assures proper sequencing of the signaling messages.

5.4.3 Level 3 – Signaling Network

This level defines the functions related to message handling and functions related to network management. The corresponding terminologies for these functions are message handling functions and signaling network management functions.

The message handling functions include discrimination, routing, and distribution of the messages. It discriminates and relays the appropriated message so that they will be rerouted to another node and send the message that is at its destination to the message distribution. The routing determines the signaling link to be used in forwarding the message that is either from the Message Distributor or from the level four of the local entity. For example, it is responsible for relaying the signaling messages along an appropriate set of STPs from one SP to the next.

The purpose of the signaling network management function is to overcome link degradation due to link failure or link congestion. It has to carry out dynamic re-routing and is responsible for monitoring the status of each link to determine the alternate routes and to recover the lost messages.

5.4.4 Level 4 – MTP User Function

The MTP defines the function and procedures for particular types of user facilities. This level consists of the ISDN User Part (ISUP), the Signaling Connection Control Part (SCCP), the Transaction Capabilities (TC), the Operations Maintenance and Administration Part (OMAP), the Telephone User Part (TUP), and the Data User Part (DUP).

The difference between the ANSI standard and CCITT standard is that TUP and DUP are not being used in the North American Telecommunication Signaling System and therefore are not included in the ANSI standard.

SCCP provides additional functions to enhance the services of the MTP. It provides means to control logical signaling connections in SS7 system, means to transfer Signaling Data Units across SS7 network, routing functions involving particular translations of messages, and management functions that control and broadcast the availability of the subsystem to the concerned nodes.

The ISUP provides the signaling functions needed to support the basic bearer services, as well as supplementary services for switched voice and non-voice applications in an ISDN environment. The ISUP can invoke the services of either the MTP or SCCP, depending on the function being performed. Prior to ISUP, the TUP was specified to provide signaling functions to support control of telephone calls on national or international connections. ISUP provides all the functions of TUP and functions in support of non-voice calls and advanced ISDN and Intelligent Network (IN) services. Some administrations have chosen TUP and others ISUP as their call control signaling protocol.

The TC contains two elements, namely Intermediate Service Part (ISP) and Transaction Capabilities Application Part (TCAP) which is divided into two sublayers called Transaction Sublayer and Component Sublayer. TC provides connectionless services to establish non-circuit-related communication between two signaling nodes. It also provides means to exchange operations and replies via a dialogue. The operations and parameters of the X.229 protocol are part of the application protocol between TC users.

AE and ASE stand for Application Entity and Application Service Element respectively. The AEs are communication functions in an Application Process, and also are sets of communication capabilities whose components are ASEs. ASE is a coherent set of integrated functions.

5.5 CCS7 Network Layout

There are two types of signaling networks, called associated network and quasi-associated network [48]. The network is associated when messages for any two SPs carried on a signaling route consists of a direct signaling link. For example, signaling point A, signaling point B, and signaling point C have direct signal links for the signaling messages. When two signaling links have a direct signaling link, the network is described as associated. Figure 5.4 (a) shows that each SP is also an exchange such that there is one trunk group associated with one signaling link (SL).

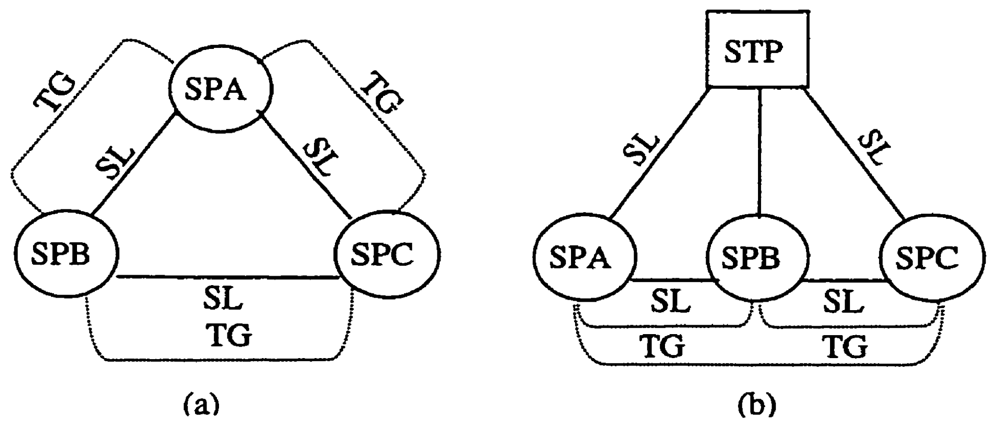


Figure 5.4: (a) Associated network (b) Quasi-associated network.

If none of the exchanges are directly connected by a signaling link and each exchange has a link to an STP, as in Figure 5.4 (b), then the network is quasi-associated. In this case, each link carries messages for several relations. All signaling routes are indirect, traversing two links, and passing through the STP.

5.5.1 Redundant Networks

A failure of a signaling link disables the signaling route(s) for which it carries CCS7 messages, and this severely affects the service in a telecommunications network. For example, if one of the signaling links is down, its associated trunk group will not be able to provide service because all the signaling messages will not be able to reach the other end of the signaling link to setup a connection for its trunk group.

The associated network in Figure 5.4 (a) has a one-to-one relationship between the SL and the trunk group. That is when the SL from SPA to SPB is down, then no calls will be established on the trunk group between SPA and SPB. The quasi-associated network has a one-to-two relationship, since it requires two signaling links to setup a connection between two signaling points. If one of the three links is down, then there are only two links left to setup the only possible connection.

Usually, redundancies are added to the network to improve the network robustness. The most common method is to add an additional signaling link to each signaling point. In the quasi-associated network, network failure can be caused by malfunction of the STP itself. Therefore, adding a redundant STP becomes a common deployment strategy. The redundant quasi-associated network is the building block of the current signaling network consisting of two STPs with multiple SPs. Each SP is connected to both STPs.

In North America, the signaling networks are divided into regions. Each region is equipped with a pair of STPs. These signaling networks provide service for different regions of telecommunications networks. The STP pairs are interconnected in a meshed fashion, as in Figure 5.5. Each SP has an A-link to the two STPs in its region. The B-links interconnect STPs of different regions, and the C-links interconnect STPs within the region.

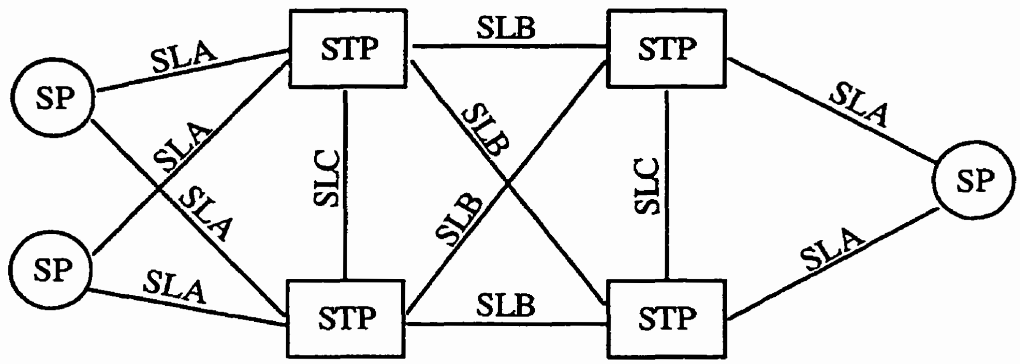


Figure 5.5: The redundant SS7 network.

5.5.2 SS7 Network Reliability

Reliability of an SS7 network can be measured in terms of availability, dependability, and robustness. Availability is defined as the mean time for the network components to fail. Its counter part, unavailability, is the mean time to repair. Frequently, system downtime has been used to measure unavailability as well.

Dependability measures the ability of a signaling network to reliably transport messages and not cause malfunctions, such as lost messages, messages out-of-sequence, transmission errors, false operations, and signaling malfunctions.

Robustness concerns the ability of a network to withstand large or catastrophic failures. The interest of network robustness is to describe how well a network is capable of withstanding major failure events. ANSI describes it as:

1. The ability of a network to maintain or restore an acceptable level of performance during failure conditions by applying various restoration techniques.

2. The mitigation of preventing service outages from potential failure conditions by applying "preventative" survivability techniques to a network.

However, there are no standard ways of measuring the two components. Another concept of robustness is the ability of a network to prevent propagation of a problem to other parts of the network.

5.6 Setting Up a Telephone Call

Setting up a connection for a local call is significantly simpler than setting up a long distance call. There are more intermediate systems in the long distance path. When a person is making a call, the call is sent to the calling party's local switch, passed to the Central Office (CO), then to the called party's local switch.

When a person is making a long distance call, the call will be passed to the local switch first, then to the CO. The call exits the CO to access a tandem which is connected to the long distance carrier network. The long distance network is also connected to the destination access tandem. Similarly, at the receiver side, the call passes through the access tandem to the CO, then to the PBX before it reaches the destination.

When calls must travel through more than one CO switch, it is necessary to transmit the called number and call status (on/off hook) from the original switch to all others involved in the call. The dialed digits and on/off hook status are called addressing and supervisory information respectively. The addressing and supervisory signals associated with the group of lines (called a trunk group) between a pair of switches are transmitted over a special data link reserved for this purpose. CCS7 is a packet switched network for signaling and supervisory data that overlays the voice path between switches. CCS7 network is faster and more economical than in-band signaling. It also allows telephone calls to be routed along different paths.

5.6.1 Basic Call Setup Message Flow

A call may or may not be successful. A successful call requires that the call goes to the *idle* line state (or *not busy*) and that the called party answers the phone. In this case, a call procedure will be processed normally [48].

In a normal situation, a completed call will involve 5 or 6 messages, depending on who hangs up first. When the origination exchange has received the complete selection information from the calling party, and has determined that the call is to be routed to another exchange, selection of a suitable, free, inter-exchange circuit takes place and an Initial Address Message (IAM) is sent to the appropriate destination. This message contains call setup information such as the called and calling numbers and the circuit identification code. IAMs are sent by each exchange until the call reaches the destination exchange.

When the destination exchange receives the IAM, it notifies the called party by sending a setup message. Normally, the called party will response with an alerting indication which is passed backwards through the network as an Address Complete Message (ACM), essentially confirming the circuit selection. When the called party answers the call, a Connect message is returned to the destination exchange, an Answer Message (ANM) is then sent to the originating exchange. A connect message is now returned to the calling party and the call setup is completed.

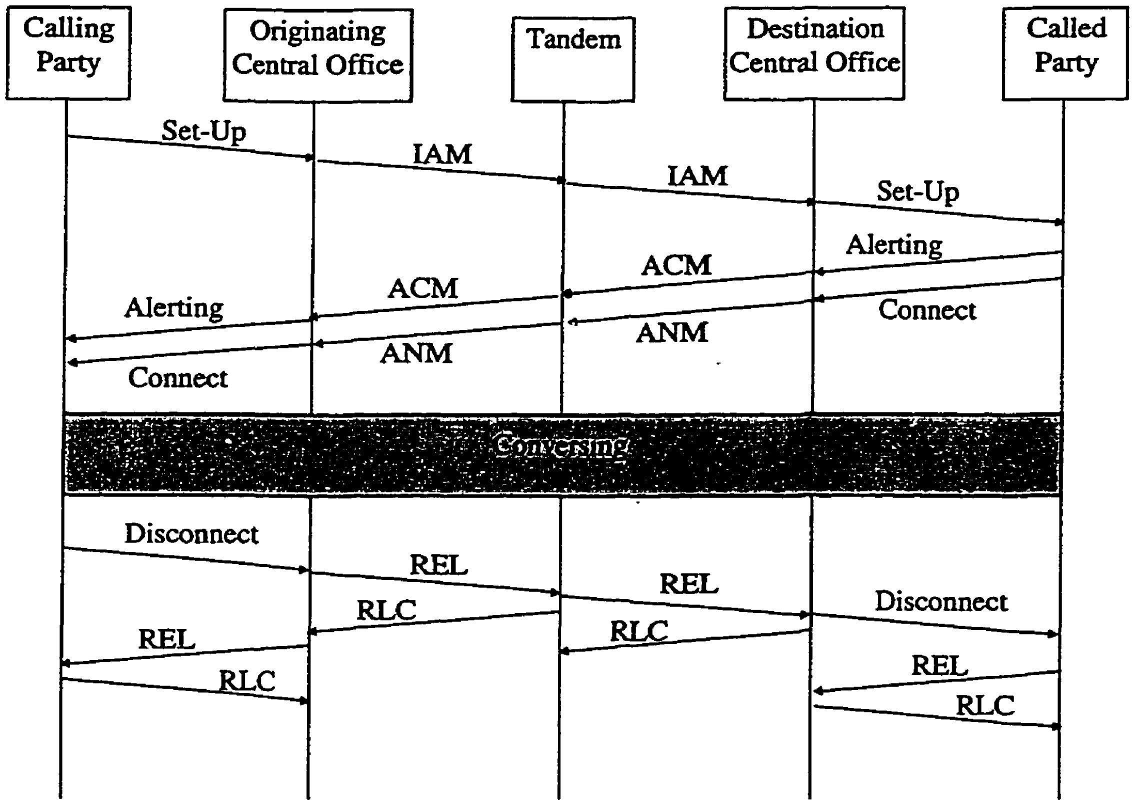


Figure 5.6: A complete call setup with calling party disconnect first.

Figure 5.6 illustrated the sequence of the message flow from one local exchange to anther local exchange, assuming that the exchanges themselves are the signaling points. There are two possible scenarios in the call disconnect procedure. If the calling party hangs up first, a Release Message (REL) is sent to the terminating office and a Release Complete (RLC) returned to the originating exchange. This procedure clears the circuit reservation, thereby allowing it to be used for another call. The second scenario is when the called party hangs up first, a Suspend Message (SUS) is sent from the destination exchange to the originating exchange. The originating exchange sends a REL and the destination exchange responds with a RLC message, clearing the trunk and finishing the processing involved with the call.

5.6.2 Other Possible Situations

Other cases like Call Busy or Call Not Answered only carry out part of the procedure. Call Busy is the situation when the called party is using the phone, and Call Not Answered is the situation when no one answered the phone and the line is free.

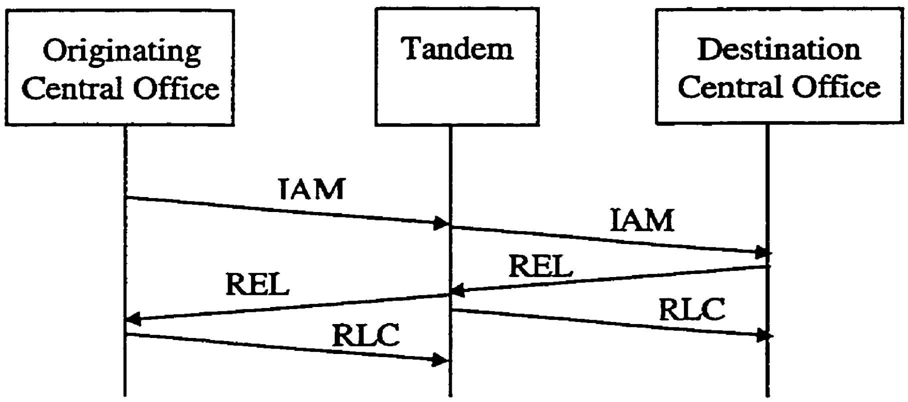


Figure 5.7: Call Busy scenario.

For the first case, when the called party is using the phone and the line is busy, only three of the signaling messages are carried out. Upon receiving the IAM from the originating exchange, the destination exchange sends back a REL message instead of an ACM and an ANM. The originating exchange frees the line, sends an RLC message, and the abnormal disconnect occurs.

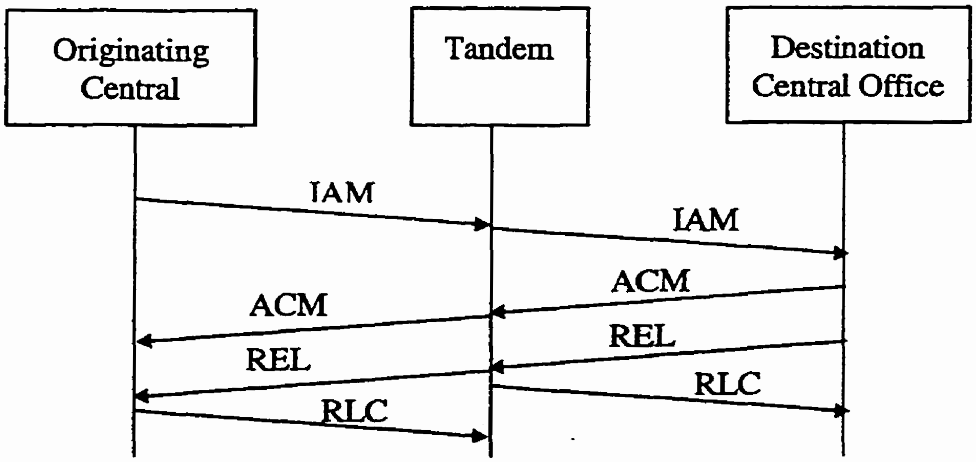


Figure 5.8: Call Not Answered scenario.

If the line is free, but the called party is not, the destination exchange sends back the ACM. When time is up and it has not received a connect message from the called party, then the destination exchange will issue a REL message and send it to the originating exchange. The originating exchange responds with the RLC message.

5.7 Local Calls

The CCS7 messages are well defined by the American National Standards Institute (ANSI) standard. The messages do not have a fixed length. The same message may vary due to different call scenarios. The purpose of the model used in the study presented here is to try to approximate the traffic characteristic of a CCS7 network.

Traffic data recorded from the MTS network was used to validate the traffic generation. The approximation method is purely iterative. In the model, source nodes are used to represent signaling messages. The packet sizes of the source nodes are modified after each trial to approximate the actual traffic load measured from the physical network.

This section and the next section demonstrate the modeling method to build CCS7 models using COMNET III and its associated protocols for value added services for telephone calls. This was a joint study with Mr. Yair Bourlas from the Manitoba Telecom Services (MTS).

5.7.1 Local Carrier Topology and Routing

A local call usually involves the two calling parties, one STP node (or called transient switch) and two SPs, the originating and the terminating node as shown in Figure 5.6. The network is modeled as the regional CCS7 network with redundant STP as shown in Figure 5.5.

The simulation model consists of two central offices where the SPs of the CCS7 network are located. Both SPs contain a set of source nodes to generate CCS7 messages. These signaling messages simulate the normal call scenario as in section 5.6.1, and two other exceptions, busy call and call-no-answer scenarios. The messages are sent from the originating central office to one of the STP nodes to setup the connection between two central offices to establish a call. The model also simulates the calling party initiating call disconnection, as illustrated in Figure 5.6.

Normally, in a redundant CCS7 network, the SPs see the links as either a primary route or secondary route. The primary route is the normal route to send the signaling messages. When the primary route has failed, the secondary route will take over the responsibility. In the COMNET III simulation environment, there is no such thing as primary route or secondary route. Instead, a routing algorithm called a minimum hop algorithm is used. The minimum hop algorithm finds the route with the least hops and uses it to deliver the data packet (signaling messages in this case). Each link is assigned with a hop value. If a route contains three links, the number of hops on that route is defined as the addition of all the hop values of its three links.

5.7.2 Calling Pattern

The number of call setups and disconnects varies during a day. The more people calling at the same time, the more chances the caller will get a busy tone. This depends on the call patterns and the number of the signaling messages that will be sent across the network. The signaling messages required to complete each scenario are generated by a sequence of messages from the source nodes of the originating and destination central offices. The name of the source nodes and scenarios are listed in the Table 5.1.

Call Type	Message Originating Node	Message Response Node	Percentage of the traffic load
Call Complete	Local-comp	LC-Complete	75%
Call Busy	Local-Busy	LC-Busy	12.5%
Call Not Answered	Local-N/A	LC-N/A	12.5%

Table 5.1: Name of the message sets and the call type table.

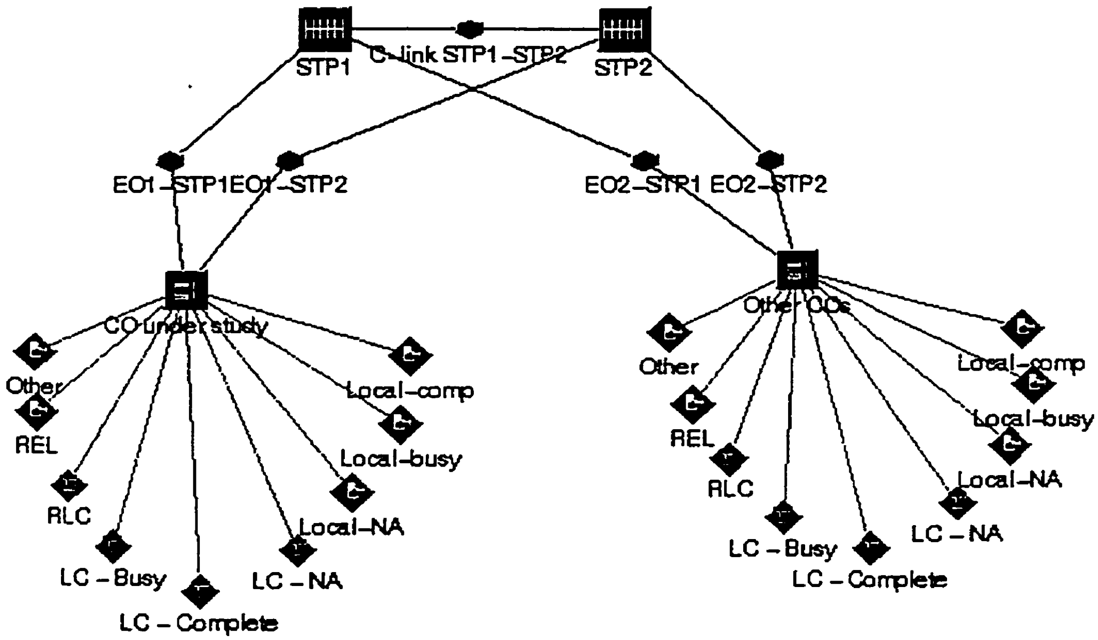


Figure 5.9: Network architecture for local calls.

5.7.2.1 Messages

After the model was built, the source nodes were configured such that they approximate the actual data measured from the physical network. Each call type was modeled as a message source with an exponential distribution. The length of the first message of every call (IAM) varies depending on the type of features invoked with the call. For example, if the call has previously been forwarded, the IAM message will contain information on the redirecting (forwarded) number.

We modeled this using a user defined distribution feature of COMNET III. As in Figure 5.9, IAM, ACM, and ANM messages were not used, instead, Local-Comp vs. LC-Complete, Local-Busy vs. LC-Busy, and Local-N/A vs. LC-N/A were used to represent each call type. Each call type contains a sequence of message flows. IAM messages varied (not uniformly) from 26 octets to 57 octets with an effective average of 52 octets. The remaining messages were of constant length, 11 octets for ACM, 9 octets for ANM, 12 octets for REL and 8 octets for RLC. Other less important messages not used in a call were grouped together and called Other as in Figure 5.9. These messages included Continuity Test (COT), Suspend, and Resume which are represented by a constant length of 10 octets in the model.

5.7.2.2 Simulation Results

The distribution of the CCS7 messages generated from the model follows the actual network very closely as in Figure 5.8.

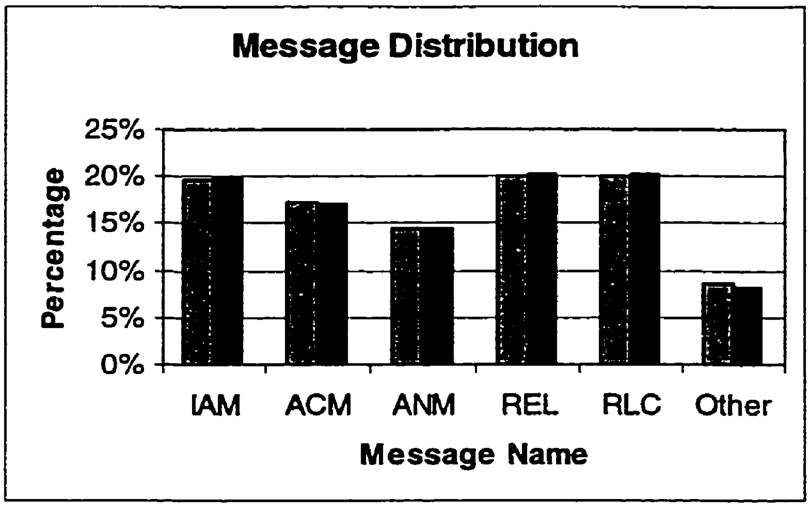


Figure 5.10: Comparison of message distribution.

The link utilization results were also agreeable with the measured data. The simulated link had, in general, 10% percent less traffic than the measured one from the physical network. Using lower traffic settings on the model mainly caused this, but the model is a good approximation to the real network. From here on, we can study a more complex network based on this simple model and the behaviours of the network under different catastrophic condition, such as one or more links are cut. This is summarized in Table 5.2.

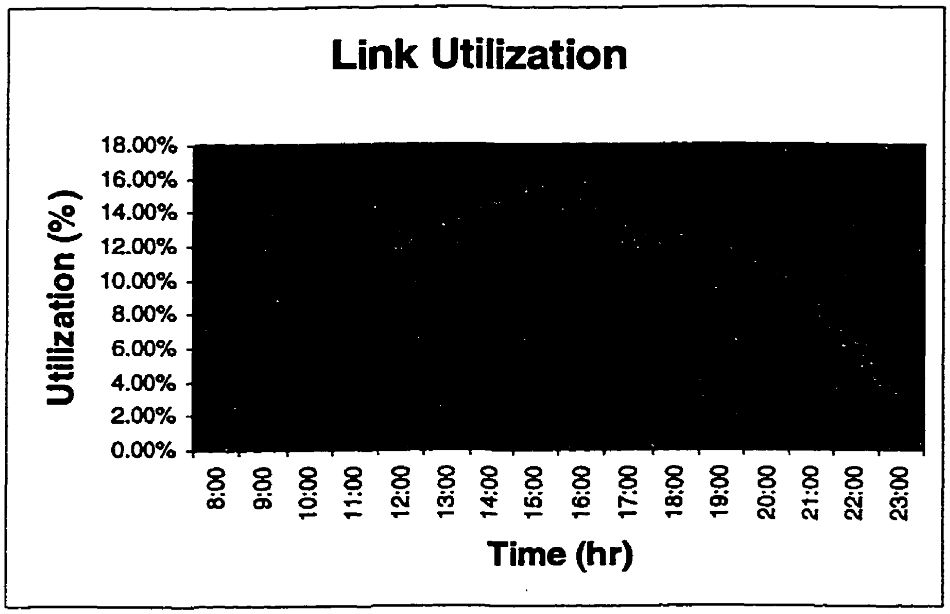


Figure 5.11: Comparison of link utilization.

	Link Utilization			Transmission Delay (ms)		
	Normal	E02-STP1 Cut	E01-STP2 & E02-STP1	Normal	E02-STP1 Cut	E01-STP2 & E02-STP1
C-link STP1->STP2	0%	0%	18.63%	0	0	1.86
Clink STP1<-STP2	0%	0%	18.06%	0	0	1.803
E01-STP1->CO	9.11%	0%	18.06%	1.818	0	1.803
E01-STP1<-CO	9.20%	0%	18.63%	1.836	0	1.86
E01-STP2->CO	8.92%	18.01%	0%	1.781	2.326	0
E01-STP2<-CO	9.46%	18.67%	0%	1.888	2.405	0
E02-STP1->Other Cos	9.20%	0%	0%	1.836	0	0
E02-STP1<-Other Cos	9.11%	0%	0%	1.818	0	0
E02-STP2->Other Cos	9.46%	18.67%	18.63	1.888	2.405	1.86
E02-STP2<-Other Cos	8.92%	18.01%	18.06	1.781	2.326	1.803

Table 5.2: Link utilization and transmission delay for one or more links.

5.8 Toll Calls and 1-800 Calls

In addition to studying different behaviours of the network based on the simple model, other features can be added to the model. For example, we can add long distance call messages and 800 call messages to the model.

A Toll call, or simply a long distance call, involves two end switches and one or more toll switches. In our study we modeled toll calls as calls that involve two toll switches and two local switches. Toll calls are separated into two general types, outgoing toll calls that originated by one of the end offices, and incoming toll calls that terminate at one of the end offices. The processes are very similar to the local calls, except that a different message will be used to represent the IAM message between the two toll offices.

The 800 calls make use of the database system at the SCP. When an 800 call is initiated, the STP or the Central Office (CO) in the network model will send a TCAP query message to the SCP to validate the 800 number. The SCP then responds with the STP to the CO about the actual long distance number. The STP will then send a toll message to the toll office to setup a long distance call.

5.8.1 CCS7 Message Sequence for Toll Calls and 800 Calls

The protocol used in toll calls and 800 calls are very much the same except the 800 calls require the toll office to query the database system to check out the actual number that is being called.

The process of making a long distance call or 800 call is the same as making a local call with more trunks in between the originator and the destination. Usually, the call has to go through the local toll office to exit the local district and the distance toll office to enter the remote district. A long distance call or 800 call can end up with one of the three common scenarios; Call Complete, Call Busy, and Call Not Answered. The COMNET III model had been expanded to cover the scenarios for the toll calls and the 800 calls.

5.8.2 Call Complete Scenarios

A completed toll call means a call had been answered and properly disconnected. Just like a local call, the local office sends an IAM to the toll office. The toll office will send the IAM message to the second toll office, and the second toll office will send the message to the remote local office. The remote office answers with the ACM and ANM messages. When the local office received both messages, the call setup is completed. Either side of the party can initiate a Disconnect message to the local office to initiate the call termination. The call is properly disconnected when the RLC message goes at both ends of the call.

Setting up an 800 call is quite similar to a toll call. However, when the toll office receives the IAM message, it will send a TCAP query message to the local database to check if the 800 number is known in the district. If the number is known, then the toll office will receive a TCAP response message with the actual calling number. The toll office will then proceed the call setup as a toll call.

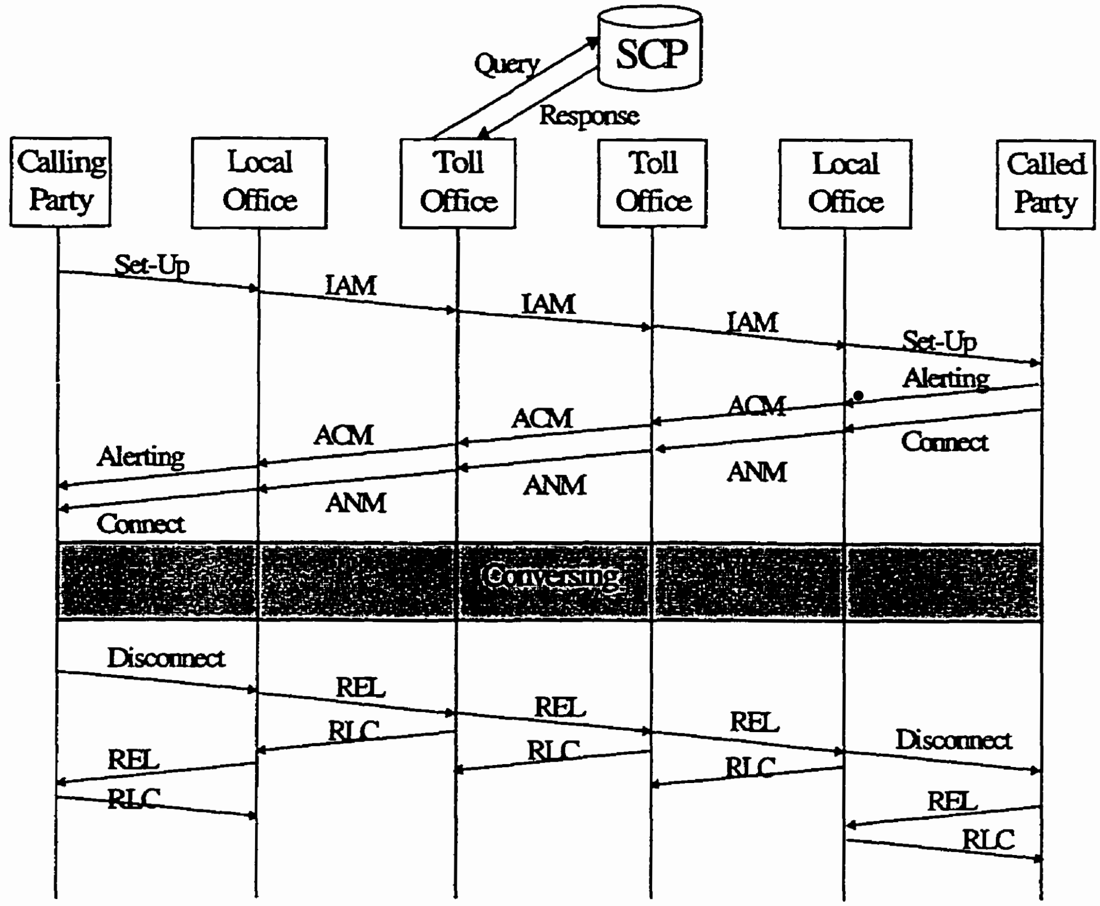


Figure 5.12: Call complete scenario for toll calls and 800 calls.

Different messages have been used to initiate different scenarios in the model. The IAM messages for the toll calls from the local office have been replaced with Toll-comp, Toll-busy, and Toll-NA respectively. Similarly, the 800 call IAM messages have also been replaced with 800-comp, 800-busy, and 800-NA. When the first toll office receives one of the 800 IAM messages, it will send a TCAP-Q message to the SCP database. The IAM messages between toll offices have also been replaced with Toll-term, Toll-term-busy, and Toll-term-NA. The completed model is illustrated as in Figure 5.13.

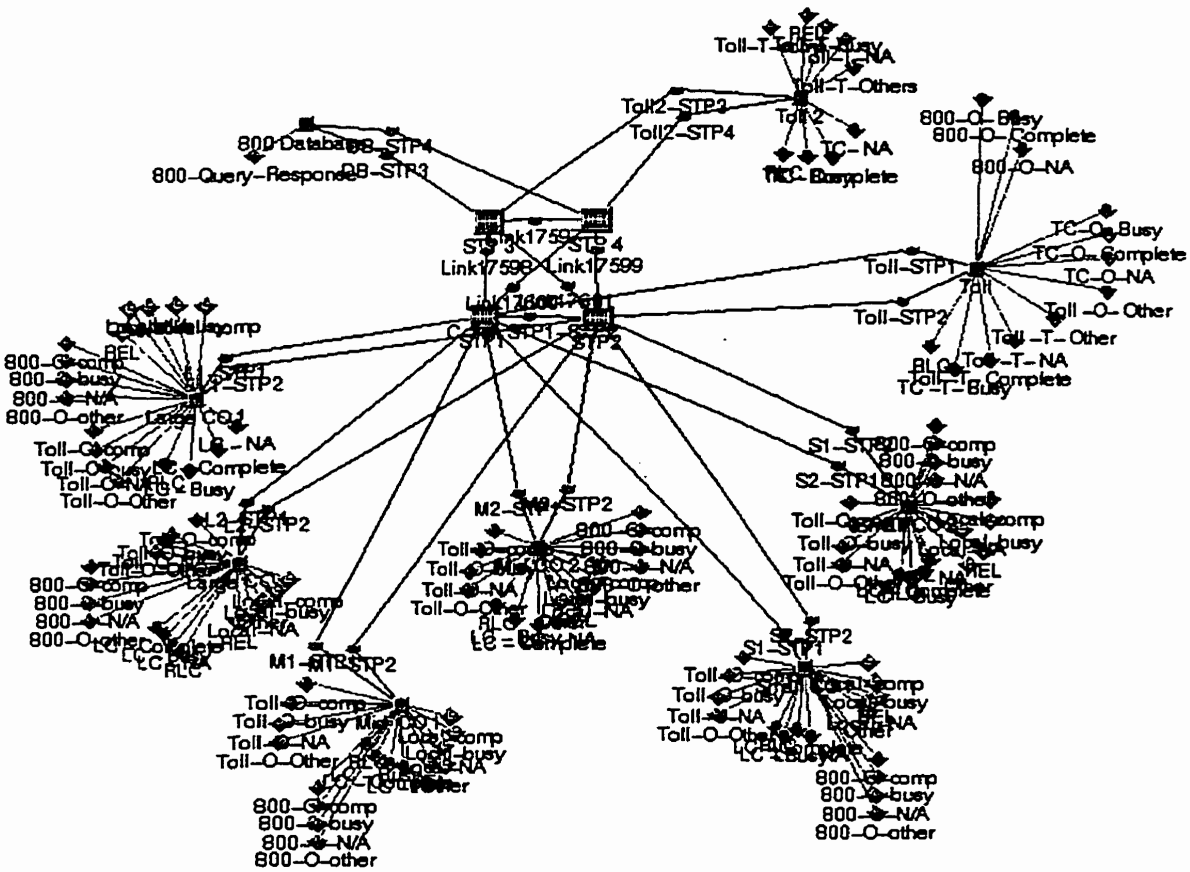


Figure 5.13: Toll/Long distance calls and 800 calls layout.

5.8.3 Impacts of the 800 Calls

The network performance can be greatly degraded or even no call can go through in a massive call event. A massive call event can be described as an event on the network when a massive amount of calls have been initiated at the same time.

For example, when someone made an advertisement that the first 50 calls get free tickets to a rock concert at a certain time. Many people will try to call at that moment. That increases the network traffic load dramatically. The associated problems to the network include link utilization, and potential processing overload at the local loop and the central office. The CCS7 network is bombarded with call setup and disconnect messages that are waiting to be processed at the STPs. It is not an easy task to model the processing power of the STP. However, if the appropriate data is available, it will be possible to determine how much network resources should be reserved for local calls and toll calls in the event of massive calls and utilize the whole network more effectively. These statistics includes the normal network utilization, the process power of the STP, the capacity of the local loop and the capacity of the central office.

5.9 Summary and Conclusions

In order to provide a better understanding of the CCS7 network, this chapter covered the foundation of all the related areas such as the ISDN network, the intelligent network, and the SS7 protocol architecture. The chapter then described the CCS7 network simulation for local calls based on the information provided by the local carrier, MTS. The model was then expanded to cover toll calls and 800 calls. However, more information will be required in order to study the impact of the massive 800 calls to the network.

Chapter 6 ATM Network Model

6.1 Asynchronous Transfer Mode

ATM technology allows asynchronous operation between the sender and the receiver. ATM technology is being selected by the International Telecommunication Union (ITU) as the switching technology for B-ISDN. ATM communication architecture is based on switching of small fixed length packet of data cells. It is also known as fast packet switching. Appendix B provides a more detailed description of the ATM technologies.

The smaller header size of ATM cells simplifies the switching and processing functionality. Unlike X.25 and frame relay, ATM technology does not perform packet retransmission, frame delimitation, or error checking. It provides a higher and wider speed range [17], and allows more variety of data to be transferred across the link.

This chapter presents an ATM network model using OPNET to approximate the MTS OC-3 network. The model was validated with traffic parameter settings and link utilization of the network. The ATM model is then modified to provide Switched Virtual Circuit (SVC) connections. The simulation result is then compared with the original

Permanent Virtual Circuit (PVC) model in terms the maximum traffic load that the network can support.

6.2 General Description of the Target ATM Network

This section describes the physical layout of the target network, services connected to the network and the amount of traffic load measured at each service connection. This section provides an introduction to the target network such as the network topology, the network services, and the network characteristics.

6.2.1 ATM Nodes and Services

The MTS ATM network comprises 11 ATM switches logically providing 2Mbps and 15Mbps of PVC connections to the customers. The switches are labeled with numeric values, numbered from 800 to 811. These switches are physically interconnected through OC-3 connections (155Mbps). The services provided to the customers are listed in Table 6.1 which shows that most of the switches are providing two or three services, and some do not provide services at all.

Service	800	802	803	804	805	806	807	808	809	810	811
Ethernet	✔					✔	✔				
WAN		✔									
ATM		✔			✔	✔	✔	✔		✔	✔
T3		✔		✔		✔	✔	✔	✔	✔	✔

Table 6.1: Services provided by the switches.

6.2.2 ATM Network Topology

Figure 6.1 illustrates the logical network layout of the ATM network. The switches are interconnected with 15Mbps and 2Mbps logical connections, in particular PVC connections. Physically, the adjacent switches are interconnected through the fiber links.

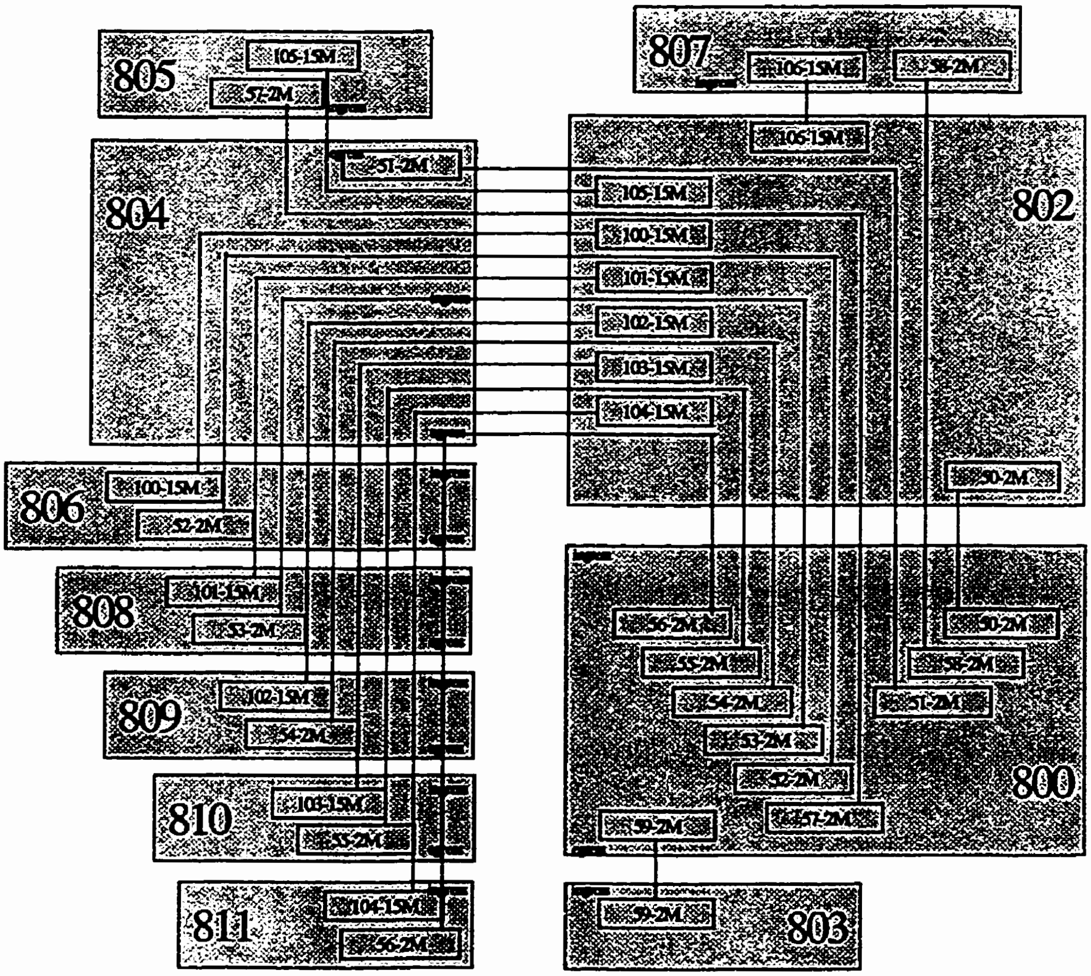


Figure 6.1: The connections between the switches.

The 15Mbps connections are labeled with a connection identifier followed by a "15M", and similarly the 2Mbps connections are labeled with "2M". The only routings in the networks are provided by switch 802 and switch 800. All the traffic will be first sent to either switch 802 or switch 800 in order to reach their destination. For instance, the 15Mbps PVC connections send all the traffic to switch 802, and switch 802 forwards the traffic to the appropriate destination. Traffic going through the 2Mbps connections will be routed at switch 800.

If customer A at switch 811 tries to communicate with customer B at switch 810 through the 15Mbps connection, the data cells will be transferred all the way to switch 802, passing through all the intermediate switches, before being routed to the destination switch.

6.2.3 Link Utilization

The way the ATM network is being setup makes the traffic go through unnecessary switches, and thus unnecessarily consume the network bandwidth. If all the switches are generating a homogeneous traffic source and send the traffic to the center switches (800 and 802), then the traffic load generated from a switch will be accumulated by the next node. The evidence is shown in the total allocated bandwidth at each link between the two switches as shown in Table 6.2.

If we treat switch 802 as the center switch, we can easily calculate the traffic load going to 802 by adding the bandwidth allocated at each link. The bandwidth allocated at switch 802 is the aggregate of the bandwidth allocated at 800⇔802, 807⇔802 and 804⇔802 links. A simply summation gives an answer of 137 Mbps which is about the limit for OC-3 rate, 149.79 Mbps of ATM traffic. But statistics show that the network resources are not being fully utilized. Figure 6.2 shows the aggregated traffic load injected to the network is less than one percent of the OC-3 rate. The network can accommodate a lot more customers.

Link	BW (Mbps)	Link	BW (Mbps)
811⇔810	17	805⇔804	17
810⇔809	34	804⇔802	102
809⇔808	51	807⇔802	17
808⇔806	68	800⇔802	20
806⇔804	85	803⇔800	2

Table 6.2: Bandwidth allocated at each switch in the direction towards switch 802.

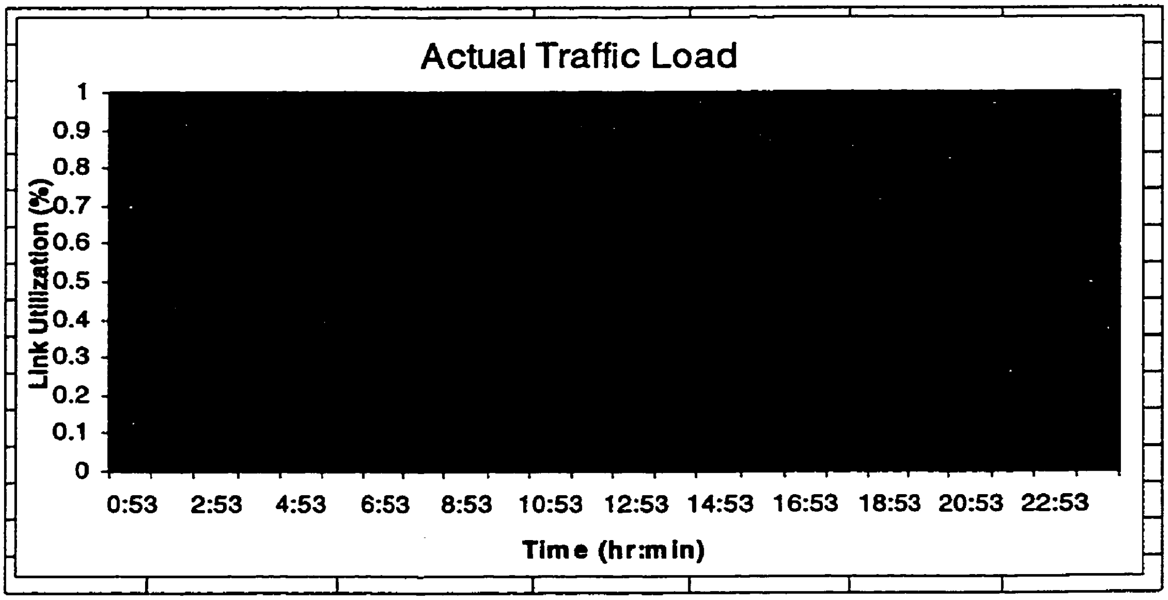


Figure 6.2: The actual link utilization.

6.3 PVC Network

The intention of the PVC network model is to study the bandwidth efficiency of the MTS network without assigning overloading on the links. Processing power and cell delay time are not considered due to insufficient information about the network. The assumptions that the model is based on are:

1. All the links connecting the ATM switches have the speed of OC-3, 155Mbps.
2. There is only one physical fiber link connecting the two switches. For example, all the connections going into switch 802 from switch 804 are carried by the same fiber link. The switch 804 multiplexes the transit and the injected ATM traffic before sending the traffic into the fiber link.
3. There are two central switches in the network, switch 800 and switch 802, and nine network access switches. The central switches are responsible for routing the cells to the appropriate switches. They also act as the receiving end-points.
4. Each network access switch contains a VCC source node and a T3 source node as illustrated in Figure 6.3. These source nodes are sending Class D ATM cells to the router switches on the network. The destination of the source nodes in switch 811, 810, 809, 808, 806, 804, 805, and 802 are set to switch 800. The source nodes in switch 803 are sending traffic destined to switch 802. The sink nodes of the other switches are not being used in this model.
5. The schedule for traffic generation is deterministic and the size of the data packet is fixed.

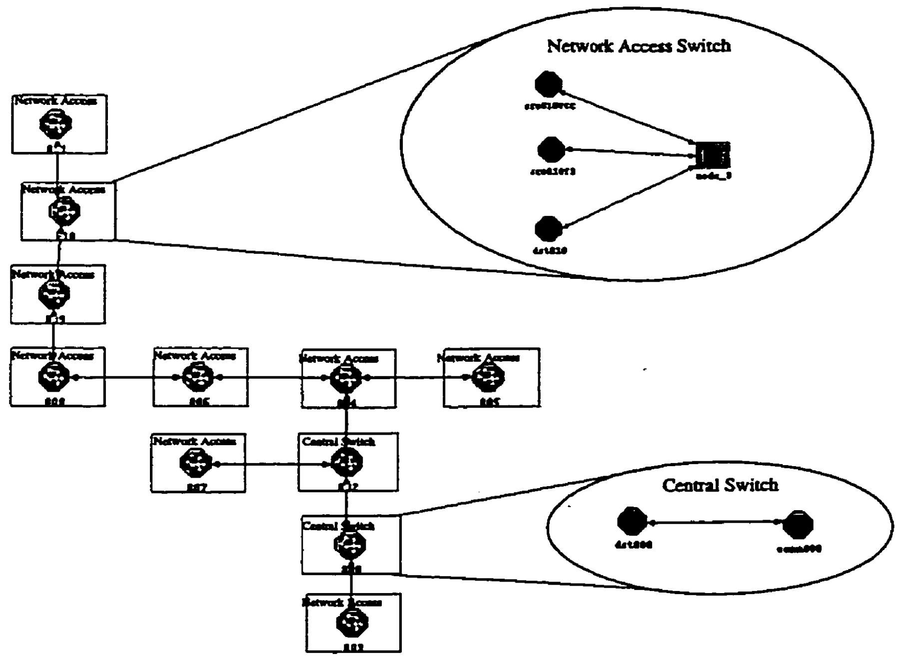


Figure 6.3: OPNET PVC network model.

6.3.1 PVC Modeling Parameters

The network is homogeneous such that all the switches contain the source nodes with the same characteristic. The source nodes of the network access switches are transmitting fixed sized Protocol Data Unit (PDU) with deterministic interarrival time.

The deterministic interarrival time is chosen for two reasons. Firstly, comparing the inconsistency of the analytical results with the simulation results can validate the network simulation. Secondly, the deterministic interarrival time can be used to simulate the worst case scenario such that the allocated bandwidth is fully utilized by the traffic load. Therefore one can imagine the connections are all carrying CBR traffic.

Each source node is generating traffic every 10 seconds and waits for another 10 seconds before generating the traffic again (on-off model). The size of the PDU is a fixed size of 1000 bits which is equivalent to 3 ATM cells with ATM Adaptation Layer type 5 (AAL 5). The network study used different interarrival times for each simulation run and the runs are 100 seconds long. The simulation time is long enough for the network to reach steady state. The values for the interarrival time used in the simulations are 0.01, 0.02, 0.05, 0.1, 0.2, and 0.5 seconds.

6.3.2 Estimate the PVC Link Utilization

Due to the nature of the traffic characteristic, the calculation for the link utilization is fairly simple. The link utilization can be calculated as a function of the interarrival time, the number of source nodes that are sending traffic across the link, the size of the burst, and the portion of the "on-time" of the source node.

If we take the interarrival time of 0.5 seconds as an example, the link utilization of link 804⇔805 is P/T multiplied by the burst size and the number of source nodes that are sending traffic through this link. In the calculation,

- P is assigned with 0.5 to represent the 50% of the time the source nodes are generating traffic.
- T is 0.5 seconds to represent the interarrival time of the packets.
- The burst size is the size of 3 ATM cells (1272 bits) in which the 1000 bits of PDU is being carried.

- The number of source nodes that utilize the link is two.

The estimated traffic rate that is passing through the link is then 2544bps. The estimated link utilization and the final record link utilization for each link are summarized in Figure 6.4 and illustrated in Table 6.3. The differences between the estimated link utilization and the final recorded average link utilization (like the convergence of the link utilization) are calculated under the title Percentage Error. There seems to be some discrepancies between the estimated and the final recorded average link utilization. This is due to the long transition time used in the simulation. There are a total of ten transitions for the source node to change from on state to off state and from off state back to on state. However, the multiplexed (or smoothed out) result at link 800⇔802 is fairly close to the estimated link utilization. The final record from the simulation is 19.825kbps as comparing to the estimated 22.896kbps.

Link	Number of Source Node	Estimated Link Utilization (bps)	Recorded Link Utilization (bps)	Percentage Error (%)
811⇔810	2	2544	3600	41.5
810⇔809	4	5088	6001	17.9
809⇔808	6	7632	8483	11.2
808⇔806	8	10176	10735	5.5
806⇔804	10	12720	12953	1.8
805⇔804	2	2544	3723	46.3
804⇔802	14	17808	17569	1.3
807⇔802	2	2544	2807	10.3
800⇔802	18	22896	19825	13.4
803⇔800	2	2544	1187	53.3

Table 6.3: PVC link utilization with interarrival time 0.5 seconds.

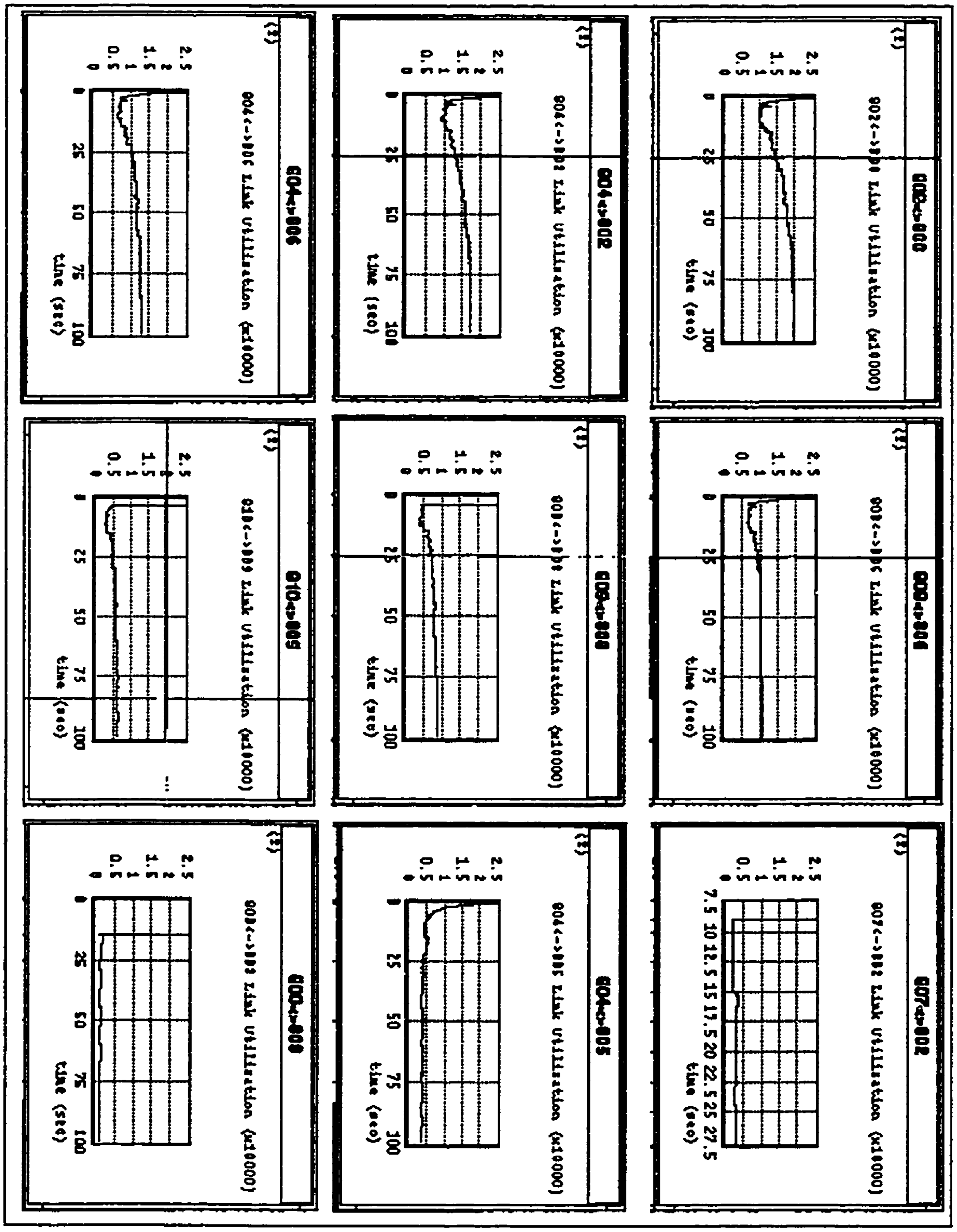


Figure 6.4: PVC link utilization with interarrival time equals 0.5 seconds.

6.3.3 Simulation of the PVC Network

The model used in this section is the same network with the same source nodes as in the last section. The interest here is to look at the link utilization of link 800⇔802 with different interarrival times.

The interarrival time values used in the simulations are 0.01, 0.02, 0.05, 0.1, 0.2, and 0.5 seconds. Along with the corresponding estimated link utilization and the final recorded link utilization, the values for the interarrival time are summarized in Table 6.4. The tracking of the simulation runs are captured in Figure 6.5. The figure and the table indicate that the simulation results are inconsistent with the estimated result with an error of approximately 20%.

Interarrival Time (sec)	Estimated Link Utilization (bps)	Simulated Link Utilization (bps)	Percentage difference (%)
0.01	1144800	889439	22.31
0.02	572400	464302	18.88
0.05	228960	187450	18.13
0.1	114480	92667	19.05
0.2	57240	46024	19.60
0.5	22896	19825	13.41

Table 6.4: PVC network using different interarrival time.

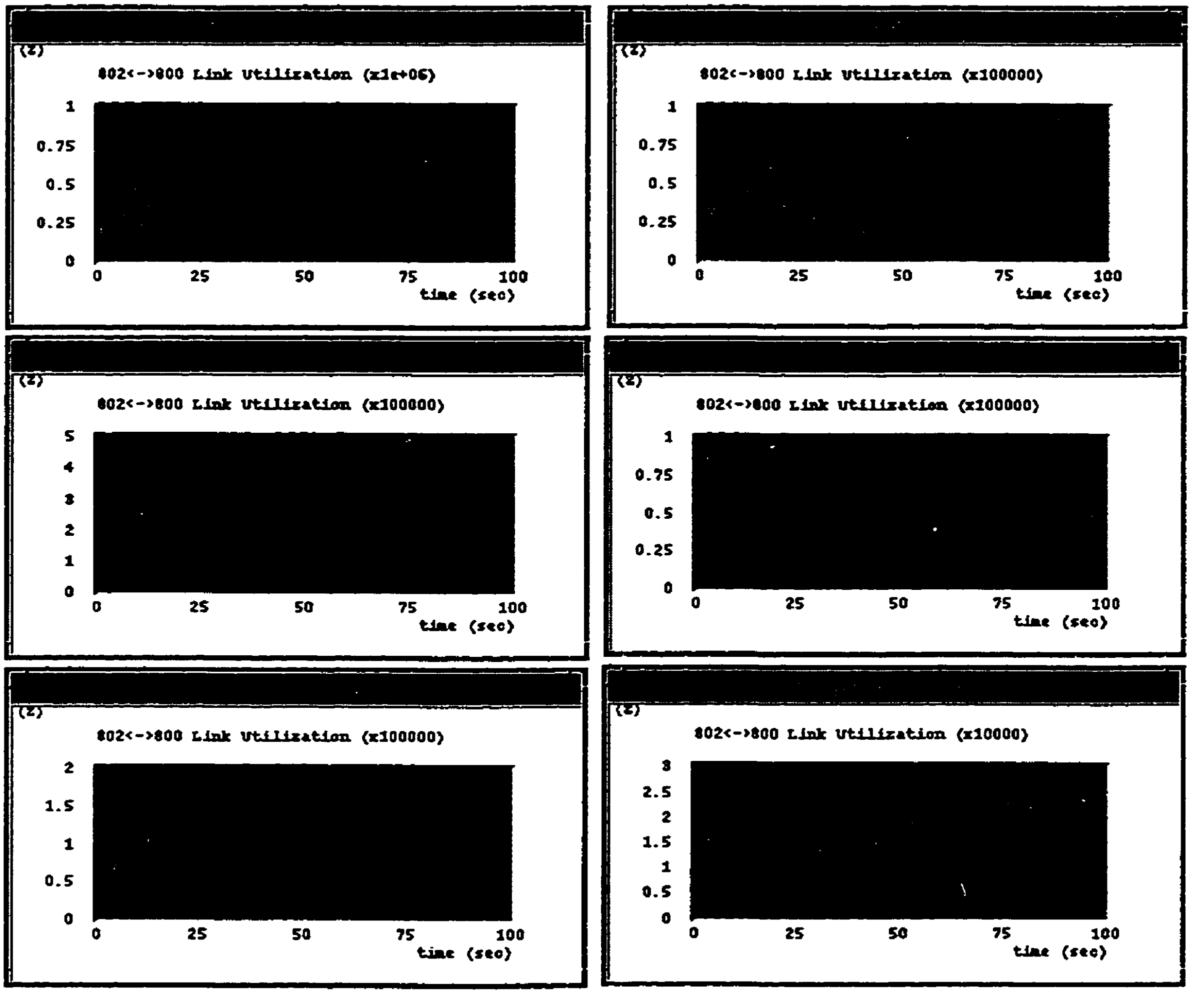


Figure 6.5: PVC link utilization for link 800⇔802.

6.4 SVC Network

This section describes the improvement in network capacity if the PVC connections have been replaced by the SVC connections. The focus here is to look at the bottlenecks of the two networks, link 800⇔802 in the PVC network and link 806⇔804 in the SVC network as demonstrated later in this section. The SVC model was built in OPNET with some modifications to handle the difference between SVC and PVC networks. The network capacity here is defined as the amount of traffic that the network can provide services to. The measurement can be easily obtained by placing a threshold on the link utilization. As seen in the last section, there is a linear relationship in the traffic load in to the network and the bottleneck of the network due to the routing scheme of the network.

The SVC network capacity can also be represented by the amount of traffic load can be added to the network to utilize the same amount of the link capacity. In the SVC network, all the SVC switches, except the destination switches, send equal amounts of traffic to all other switches. The network is modeled as follows:

1. All the links connecting the ATM switches have the speed of OC-3, 155Mbps.
2. There is only one physical fiber link connecting the two switches.
3. There are nine source/destination switches with no more center switches. The center switches had been replaced with the destination switches, switch 800 and switch 802, where the other network access switches can send traffic to them.
4. Each source switch contains 10 source nodes and a destination node. Each source node is sending data only to its destination node in a switch.
5. The schedule for traffic generation is deterministic and the size of the data packet is fixed.

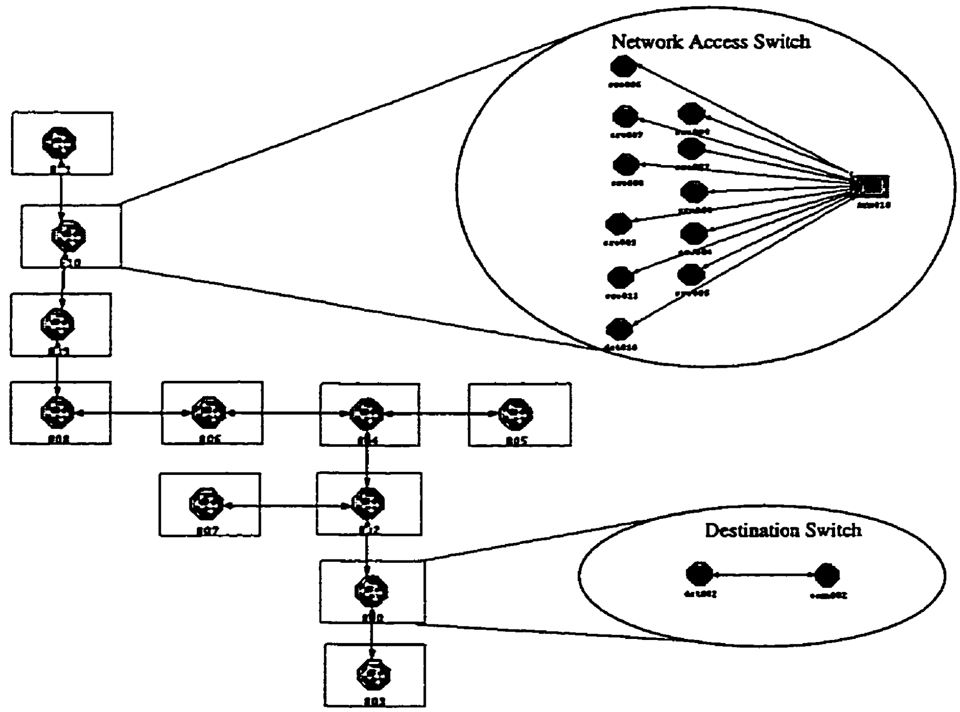


Figure 6.6: OPNET SVC network model.

6.4.1 SVC Modeling Parameters

The parameters used in the SVC model are the same as the ones used in the PVC model. The one thousand bits of PDU is encapsulated in the AAL5 format and segmented into three ATM cells before being sent out to the network. The source nodes are based on On/Off model which start generating for 10 seconds and stop transmitting for 10 seconds before generating traffic again.

6.4.2 The Estimated SVC link Utilization

In order to calculate the link utilization for this model, one has to know the number of source nodes that are sending traffic across the link. If one assumes that the amount of traffic generated from a source node as one unit using interarrival time 0.05 seconds, T=0.05. Counting the number of traffic units that the link is supporting multiplied by the unit traffic quantity will give us the traffic load.

Figure 6.7 is a logical layout of the network labeled with the traffic units on each link. The number of traffic units passing through link 811⇔810 can be determined as 10 units, generated from the switch 811, and 8 units, coming from the other 8 switches. Adding the number of the traffic units for both directions together will obtain the total traffic units. The actual traffic load can be calculated as P/T multiplied with the traffic units and packet size with P=0.5. Thus, the link utilization on link 811⇔810 is 228,960 bps.

The number of the traffic units passing through link 810⇔809 can be determined as 9 units, from switch 810 and switch 811 sending to the other 9 switches and 7 units generated from other source nodes sending traffic to switch 810 and switch 811. The link utilization on link 810⇔809 is 407,040 bps. Similarly, other link utilization can be calculated using the same method. The number of traffic units on each link, the estimated link utilization, and the final recorded link utilization from the simulation are summarized in Table 6.5. The table indicates that there are 10% differences in the two link utilization values. However, the estimated and simulated results are within the same bold part, and both results agree that link 806⇔804 is the bottleneck of the network.

Link	Up/ Right traffic units	Down/ Left traffic units	Total traffic units	Estimated link utilization	Recorded link utilization	Percentage difference (%)
811⇔810	10	8	18	228960	196524	14.17
810⇔809	18	14	32	407040	356949	12.31
809⇔808	24	18	42	534240	486108	9.01
808⇔806	28	20	48	610560	545022	10.73
806⇔804	30	20	50	636000	563361	11.42
805⇔804	8	10	18	228960	195371	14.67
804⇔802	28	14	42	534240	441688	17.32
807⇔802	10	8	18	228960	230026	0.47
800⇔802	16	9	25	318000	307341	3.35
803⇔800	8	10	18	228960	227557	0.61

Table 6.5: SVC link utilization with interarrival time 0.05 seconds.

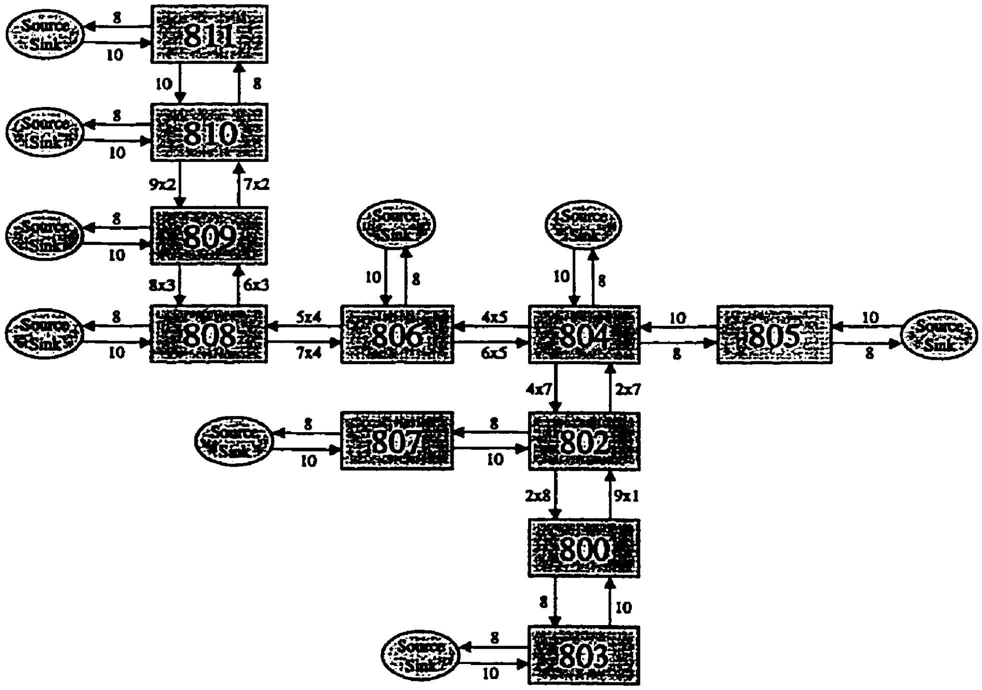


Figure 6.7: Units of traffic passing through a link.

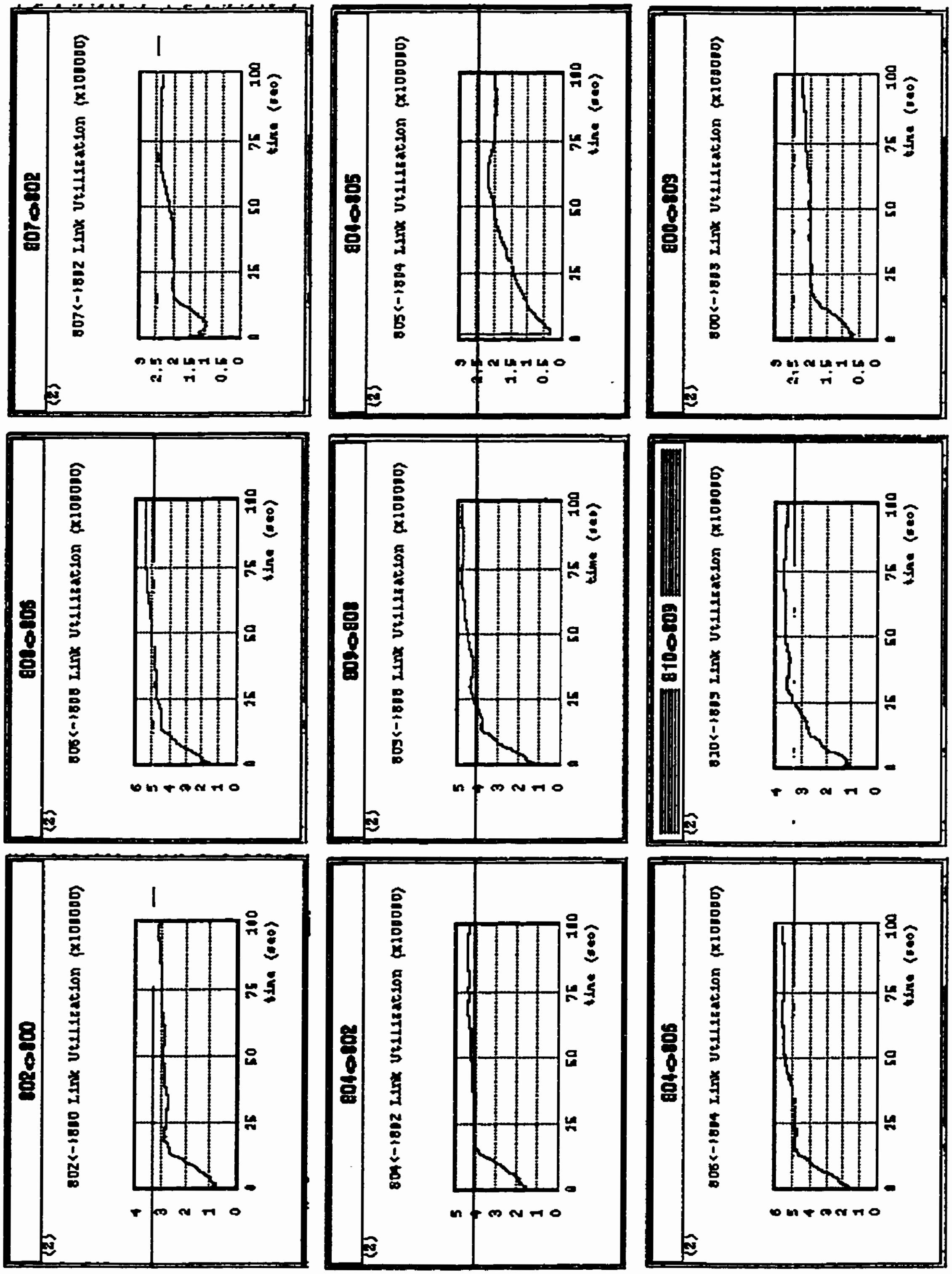


Figure 6.8: SVC link utilization with interarrival time 0.05 seconds.

6.4.3 Simulation of the SVC Network

The model had been simulated 12 times using different interarrival times to study the potential congestion on the network, namely link 806⇔804. These interarrival values are 0.02, 0.03, 0.04, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, and 50 seconds.

The interesting finding is the great difference between the estimated result and the final link utilization when the traffic generation rate decreases. When the traffic is being generated at such a slow rate, the model required more simulation time for the traffic generation to reach the steady-state for time averaging. There is a significant transition point on the two link utilization columns in Table 6.6. The final link utilization is usually below the estimated link utilization. As the interarrival time increases, the final link utilization reaches a point where the result exceeds the estimated link utilization. The difference between the two grows as the interarrival time increases.

Interarrival Time (sec)	Estimated Link Utilization (bps)	Final Link Utilization (bps)	Percentage difference (%)
0.02	1590000	1369457	13.87
0.03	1060000	792199	25.264
0.04	795000	748727	5.82
0.05	636000	563361	11.42
0.1	318000	272565	14.28
0.2	159000	151080	4.98
0.5	63600	69091	8.63
1	31800	39450	24.06
2	15900	26327	65.58
5	6360	18438	189.9
10	3180	16264	411.44
50	636	13609	2039.78

Table 6.6: PVC network using different interarrival time.

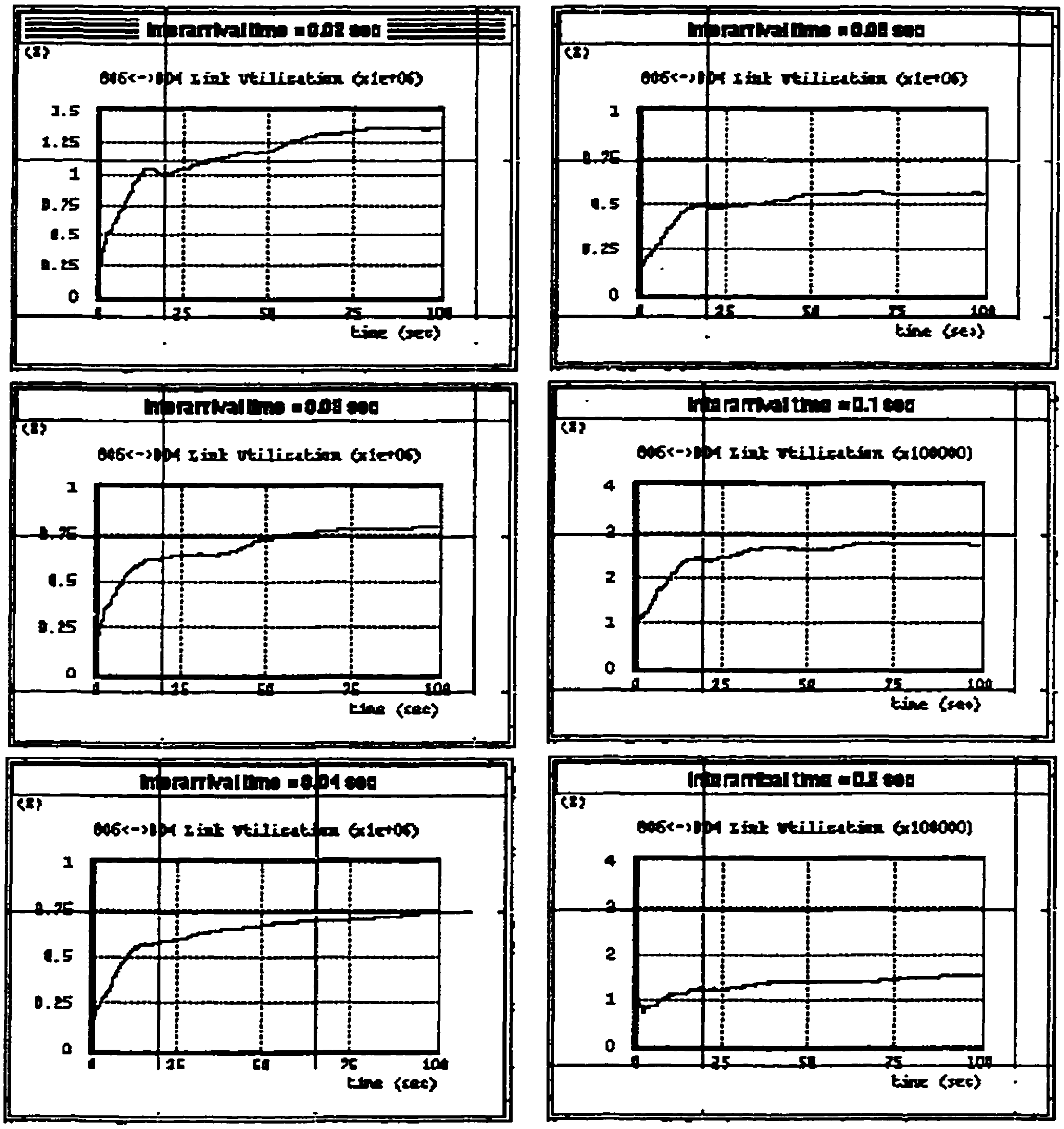


Figure 6.9: SVC link utilization for link 806⇔804 with interarrival time 0.02, 0.03, 0.04, 0.05, 0.2, and 0.2 seconds.

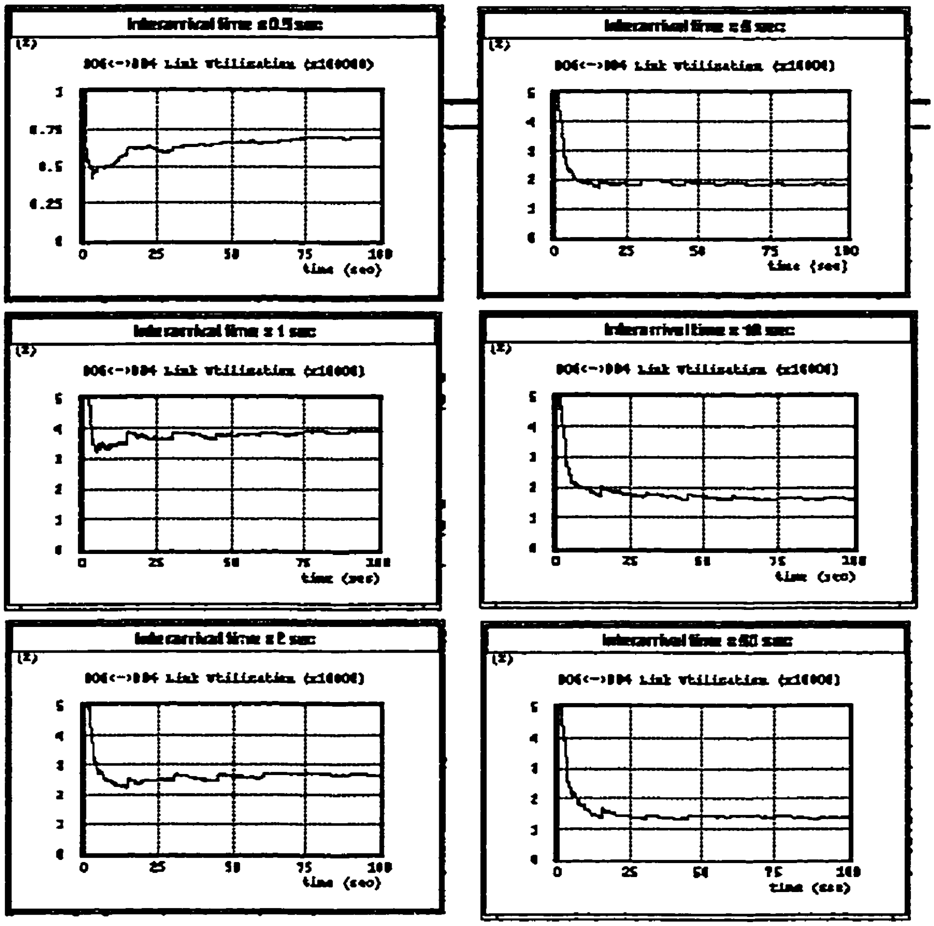


Figure 6.10: SVC link utilization for link 806⇔804 with interarrival time 0.5, 1, 2, 5, 10, and 50 seconds.

6.5 Comparing the Two Networks

One of the limitations on how much service a network can provide is its link capacity, and also how the network handles the traffic to utilize its limited resources. In this chapter, the network has been modeled as a PVC network and a SVC network. There is a significant difference between the link utilization from the figures shown in Table 6.4 and Table 6.6. The difference in the network capacity can be calculated by simply comparing the amount of traffic going through the bottleneck of the networks, namely link 802⇔800 in the PVC network and link 806⇔804 in the SVC network.

The estimated results indicate the SVC network can handle 2.77 times more traffic than the PVC network. The simulation results say the difference between the two is around 3 times. The SVC network significantly increases the network capacity to handle more traffic. However, one has to consider the excessive processing to provide SVC types of services. One has to trade-off the price differences between laying out the fat pipes for the PVC network to get the equal capacity and buying a super-processor and the right software package to provide the SVC services.

Chapter 7 ATM Congestion Control: an Original Approach

7.1 Introduction

This chapter presents an original link-by-link congestion control mechanism using a fuzzy inference method. The method utilizes the linguistic ability of fuzzy set theory and logic to control the injection rate of the Available Bit Rate (ABR) traffic sent by each preceding Asynchronous Transfer Mode (ATM) switch. The complexity of the fuzzy inference method makes it very difficult to perform real-time computations, and for this reason we have approximated the controller using both a fuzzy neural network (FNN) and an artificial neural network (ANN), and we have studied how well they approximate it. Supervised learning is the learning algorithm through which the FNN and ANN approximate the inference method. This chapter shows that ANN is a preferable implementation of the controller.

The new approach for congestion avoidance in ATM presented in this chapter can be built in OPNET as process models. A feature of OPNET that is not in COMNET III is the ability to import user source code into the model. An algorithm written in C, can be put directly into an OPNET model to test how well the algorithm performs. It is desirable to evaluate the performance of new components in simulation and study their impact on a network, before they are put into the real network. The simulations presented in this chapter were built in C with the objective of making the code available for OPNET simulations.

7.2 ATM Congestion Control Mechanisms Overview

ATM network is a statistics-based non-deterministic traffic transportation system. The switching is done by statistically multiplexing several connections on the same link based on their traffic characteristics. The statistical behaviour of the traffic in ATM networks is characterized by peak-bit rate, average-bit rate, burst length, and even more complicated second order time domain parameters, such as Index of Dispersion for Counts (IDC) [35]. However, congestion may happen when multiple bursty traffic streams occur simultaneously.

ATM networks take the advantage of the burstiness of traffic via the use of statistical multiplexing. A call can potentially exceed the negotiated capacity, and for this reason there is a need for traffic congestion control functions. An ATM traffic control system is a set of actions taken by the network management system to avoid congestion conditions or to minimize congestion effects. The functions include:

- Network Resource Management (NRM),
- Connection Admission Control (CAC),

- Usage Parameter Control (UPC) or Policing Function (PF),
- Priority Shaping (PS),
- Fast resource management.

In practice, even with all the functions implemented, an ATM network cannot guarantee to provide the negotiated (Quality of Service) QoS [42]. In the past few years, considerable research has been focused in the area of resource allocation, policing metering, shaping misbehaving traffic, congestion avoidance and control in ATM networks. The common objective is to find a solution that can solve the intricate congestion problem. The methodology presented here is an attempt to better utilize the limited resources by metering and controlling the injection rate of the traffic from a previous switch in the traffic path using fuzzy set theory and fuzzy logic.

7.3 Fuzzy Logic-Based Congestion Control

The goal here is to maximize the bandwidth utilization, and at the same time avoid any possible congestion situation, by controlling the Available Bit Rate (ABR) traffic using a feedback notification method.

If more bandwidth is available, the preceding switches will be informed to increase the ABR injection rate in response to the changes in Variable Bit Rate (VBR) traffic. The new controller is responsible for issuing a signal to the previous switches to adjust the ABR traffic injection rate. The signal carries the information about the need to increase, decrease, or maintain the injection rate by monitoring the traffic load and queue size of the buffer. The importance of using fuzzy set theory is that it can provide a robust mathematical-logical framework to handle problems of imprecision and statistical uncertainty. The advantage of using neural networks is that the algorithms are less complicated than fuzzy inference methods, and thus require less computation time.

The controller is intended to be used on line, and it will generate a signal for every 20 ATM cell arrivals. The fuzzy inference method makes decisions based on the buffer availability B and the current traffic rate, represented by the cell interarrival time T. The FNN and the ANN will map their input parameters, T and B, to an output signal as in the inference method.

7.4 Membership Functions

The membership functions used in the FNN and the ANN are the same as the ones in the inference method as described in this section. The membership functions are unity for full membership see Figure 7.1.

7.4.1 The Time Variable (T)

The first input considered is the normalized mean interarrival times of 20 cells. The interarrival time can determine the traffic condition of a switch. The normalized range for interarrival time is converted into the universe of discourse $T \in [0,1]$. The time limit is set to be the minimum time required to switch four cells. Its value for a 155 Megabit/sec switch is 11 μs. Any calculated mean interarrival time longer than the time limit will be set to maximum. Three continuous Gaussian shaped membership functions, leveling off to the left and right, are defined for the traffic conditions. The fuzzy sets are defined as

$$mT = \{mTH(T), mTM(T), mTL(T)\}, \text{ where } mT \in [0,1]. \tag{7. 1}$$

These membership functions are representing Heavy (*mTH*) , Medium (*mTM*), and Light (*mTL*) traffic. The Gaussian shape membership functions are:

$$\begin{aligned} mTH(T) &= \begin{cases} 1, & 0 \le T \le 0.25 \\ e^{-(T-0.25)^2}, & 0.25 < T \le 1 \end{cases} \\ mTM(T) &= \left\{ e^{-(T-0.5)^2}, \quad 0 \le T \le 1 \right. \\ mTL(T) &= \begin{cases} e^{-(Tt-0.75)^2}, & 0 \le T \le 0.75 \\ 1, & 0 \le T \le 1 \end{cases} \end{aligned} \tag{7.2}$$

7.4.2 The Buffer Variable (*B*)

The second input is the normalized available buffer size in the universe of discourse $B \in [0,1]$. This input is the unused buffer normalized to the size of the buffer. Two piecewise linear functions are assigned for the membership functions, and they represent when the buffer level is High (*mBH*) or Low (*mBL*). The corresponding fuzzy sets are defined as

$$mB = \{mBH(B), mBL(B)\}, \text{ where } mB \in [0,1]. \tag{7.3}$$

The membership functions compliment each other, the buffer availability is either high or low. The corresponding membership functions for *mBH(B)* and *mBL(B)* are:

$$
mBH(B)=\begin{cases} 1, & 0 \le B < 0.3 \\ -2.5B+1.75, & 0.3 \le B < 0.7 \\ 0, & 0.7 \le B \le 1 \end{cases}
$$
$$
mBL(B)=\begin{cases} 0, & 0 \le B < 0.3 \\ 2.5B-0.75, & 0.3 \le B < 0.7 \\ 1, & 0.7 \le B \le 1 \end{cases} \tag{7.4}
$$

7.4.3 The Output Variable (*O*)

Similarly, the output is classified into three fuzzy sets in the universe of discourse $O \in [0,1]$. The membership function of the output fuzzy sets are non-overlapping triangle-shape functions, leveling off to the left and right. We define the output fuzzy set as

$$
mO = \{R(O), N(O), I(O)\} \text{ where } mO \in [0,1]. \tag{7.5}
$$

The membership functions *R, N,* and *I* are representing the output sets Reduce, No-change, or Increase, respectively. The output membership functions are also piece-wise linear with the corners at O={0.2, 0.35, 0.5, 0.65, 0.8} as shown in Figures 7.1 to 7.3.

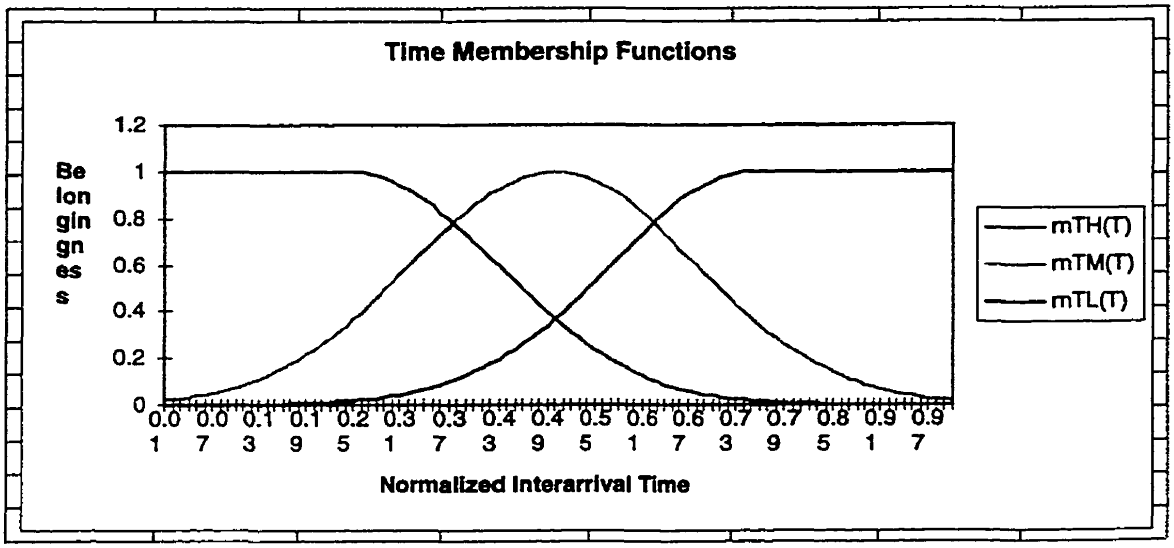


Figure 7.1: Membership functions for time *T*.

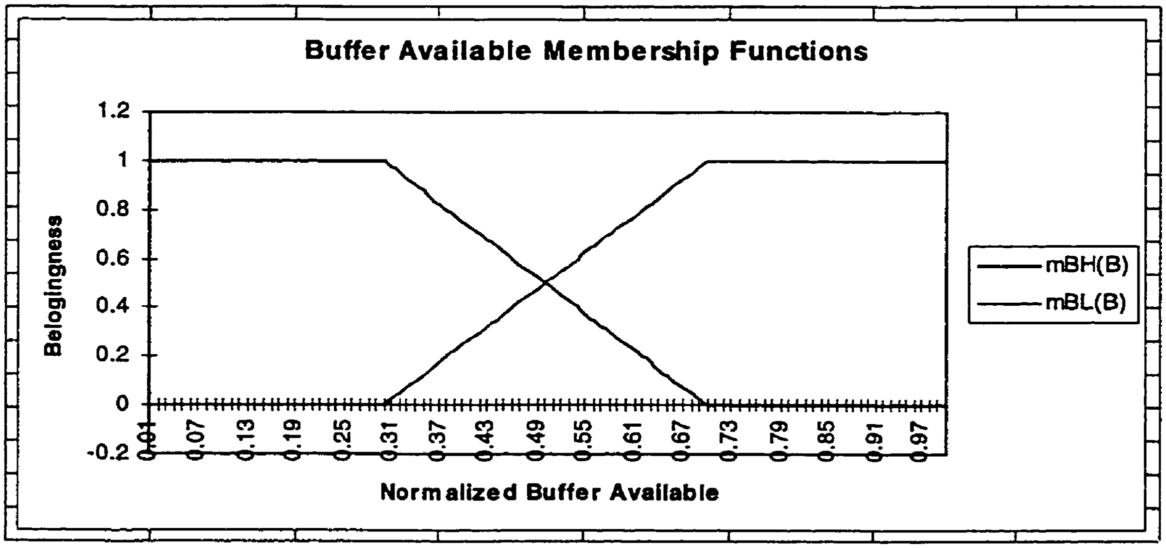


Figure 7.2: Membership functions for available buffer B.

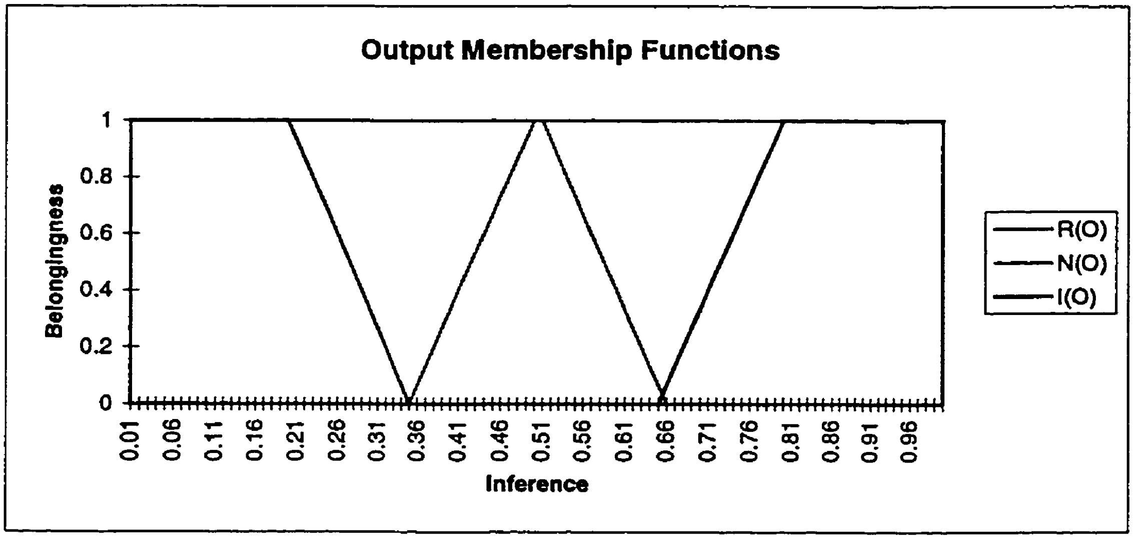


Figure 7.3: Output membership functions.

7.5 Fuzzy Rules

The two-input, single-output inference method is composed of six two-dimensional rules [36] [41]. The compatibility for each of the antecedent conditions of the rules and the input is defined as the minimum of the two. The output is assigned to the appropriate class based on these rules. The rules can be described as a set of conditional if-then statements:

Rule 1: if T is mTH and B is mBH, then O is R

Rule 2: if T is mTM and B is mBH, then O is R

Rule 3: if T is mTL and B is mBH, then O is N

Rule 4: if T is mTH and B is mBL, then O is R

Rule 5: if T is mTM and B is mBL, then O is N

Rule 6: if T is mTL and B is mBL, then O is I

T, B and O are linguistic variables, and will be referred to as antecedents and result of inference. mT, mB, and mO are the linguistic values of the inputs and outputs, that is the belongingness to the membership functions. The inference method is a fuzzy rule-based system, which will make the decision to reduce, increase, or perform no change on the traffic for each set of antecedents T and B. The antecedents are mapped into their membership values, and the minimum of these two values in each rule will be part of the contribution to define the overall inference. The above rules are summarized in Table 7.1.

	mTH	*mTM*	*mTL*
mBH	*R*	*R*	*N*
mBL	*R*	*N*	*I*

Table 7.1: Fuzzy rules.

The overall inference is defined as the centroid, that is

$$O^* = \frac{\int mO^*(O)OdO}{\int mO^*(O)dO} \tag{7.6}$$

The function $mO^*(O)$ is constructed from the inference result of each rule, and the minimum value is taken into account. The outputs of the neurons are the fuzzy measurements of the overall inference for the output sets.

7.6 System Architecture

The controller system is constructed of the following four modules: function module, network system module, decision-making module, and inference method module, as shown in Figure 7.4. In the case of FNN controller, the function module is called the fuzzification module. Mapping of the inputs into fuzzy sets is performed in the fuzzification module, where the input parameters interarrival time (*T*) and available buffer size (*B*) are mapped into their corresponding fuzzy sets. The network module is the FNN with built-in defuzzification at the output layer. Figure 7.5 illustrates the systems of the FNN and ANN network modules. In the ANN controller, the function module is a mapping, which maps the inputs, *T* and *B*, into different values. The functions that are used to map the inputs are the same functions used in the fuzzification module of the

FNN controller. The network module is a feedforward ANN with three perceptrons. The decision-making module makes the decision of sending the appropriate signal. The inference method module is only used during training and testing of the controller, and its role is to supervise the controllers. The inference module and the fuzzification module make up a fuzzy rule-based system. The fuzzy rule-based system combines fuzzy sets and linguistic variables with rules and inferencing. At the end of testing, one controller, as in Figure 7.4 without the inference module, will be used to control the injection rate of the ABR traffic.

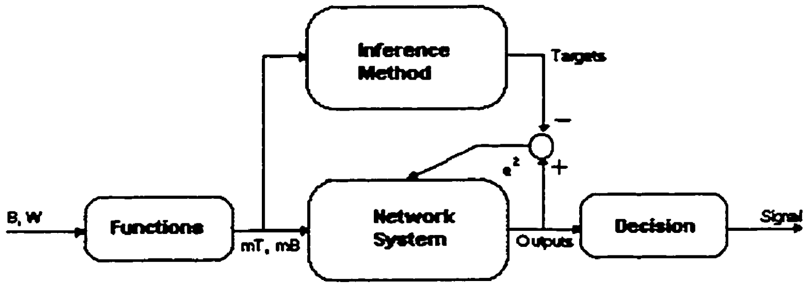


Figure 7.4: Controller architecture.

7.6.1 The Fuzzy Neural Network

The FNN network module consists of a feedforward fuzzy network with a single hidden layer and an output layer. The inputs to the FNN are the membership values of the fuzzy set. The hidden layer neurons carry a set of fuzzy rules as in the inference method. Each element in the hidden layer is fully connected to the output layer. The inference results of the fuzzy rules are transferred to the output layer. The output layer contains three AND fuzzy neurons, which compute the overall inference result and defuzzifies it into three outputs. We will call these fuzzy neurons Reduce neuron (*FNN_R*), No-change neuron (*FNN_N*), and Increase neuron (*FNN_I*). The outputs, *FNN_O = {FNN_R, FNN_N, FNN_I}*, of the neurons represent the intensities of the output fuzzy sets as described in Section 7.3. The desired outputs of the network module are based on a set of fuzzy rules and the connections connecting the hidden layer to the output layer for a certain traffic load and queue length. The AND fuzzy neurons are defined by a pair of t-norm and s-norm operators.

$$AND(X, F\_w) = \mathrm{t}_{i=1}^{n}\left(X_i \,\mathrm{s}\, F\_w_i\right) \tag{7.7}$$

$$\text{t-norm: } X \,\mathrm{t}\, F\_w = XF\_w \tag{7.8}$$

$$\text{s-norm: } X \,\mathrm{s}\, F\_w = X + F\_w - XF\_w \tag{7.9}$$

The X and F_w are vectors with six entities, and they are the inference result of the fuzzy rules at the hidden layer and their weight connecting the output neurons. The value of n is 3. This corresponds to the reduce, no-change and increase neurons.

During the training process, the outputs of the FNN are compared with the desired target values, and a set of errors is generated. The square errors will be fed back to the FNN to adjust the weights of the connections using a gradient-descent method. The target values are generated by an inference module which computes the centroid, or "center of mass". The centroid determines the corresponding belongingness of the output fuzzy sets. The decision module makes the final decision based on the output intensity of each neuron; the one with the greatest value wins. The inference module computes the conventional inference method and generates the corresponding output belongingness for each output membership function.

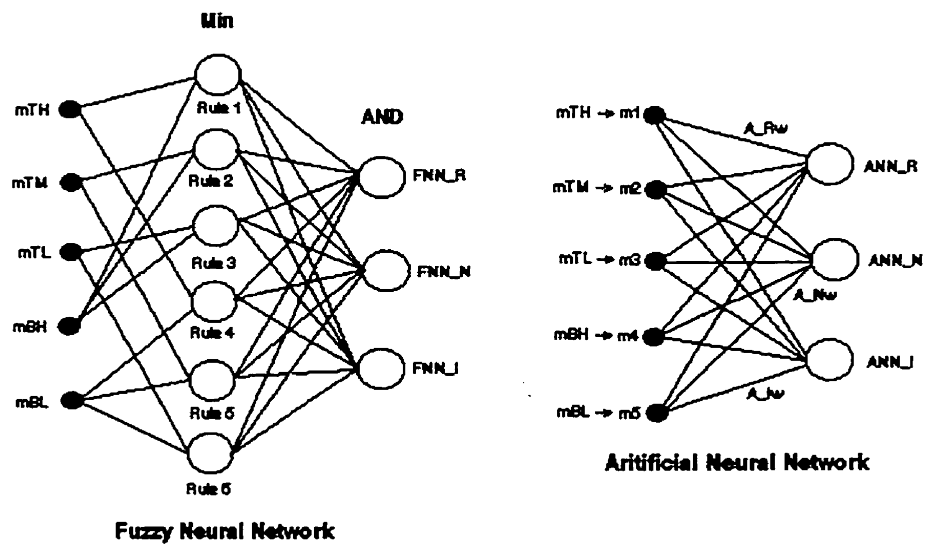


Figure 7.5: The FNN and ANN network modules.

7.6.2 Artificial Neural Network

The ANN is a single layer network, and it is constructed of three simple perceptrons. Each perceptron follows a fuzzy value calculated by the inference method. The inputs to the ANN are the results of the mapping function previously discussed. The functions used in mapping are the same as the membership functions in the FNN, see Section 7.3. We will use m = {*mTH, mTM, mTL, mBH, mBL*} to represent the input layer. The three perceptrons at the output layer are called linear units. The output values, *ANN_O* = {*ANN_R, ANN_N, ANN_I*}, are the weighted sum of the inputs. The sets of weights connecting the input and the output layers will be denoted as a set of vector A_w = {*A_Rw, A_Nw, A_Iw*}.

The ANN training process is similar to FNN training process. The connection weights are adjusted during training with a gradient-descent method. The ANN outputs are compared with the target values generated by the inference method, and a set of errors will be fed back to the ANN to adjust interconnection weights. The decision module chooses the winner with the greatest output value.

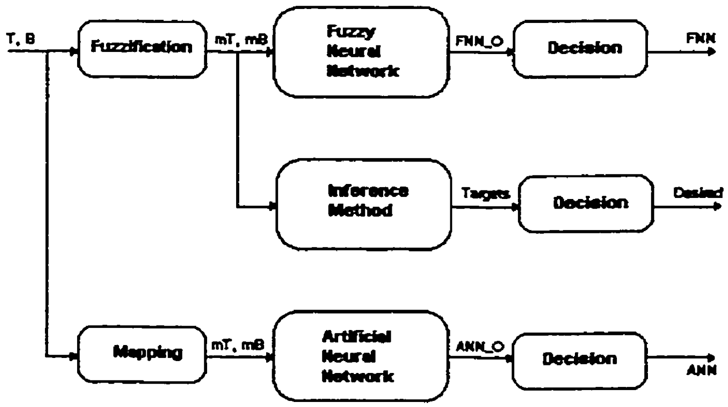


Figure 7.6: The testing process.

During the testing process, both controllers and the inference module were simultaneously fed with the same set of input data as in Figure 7.6. The final decisions and the time required to carry out the task will be recorded and compared.

7.7 The Learning Algorithm

The learning for both networks is done through a backpropagation algorithm with a square error as the performance index, also called the cost function, as in Equation 7.11. The square error is propagated back to modify the connections of the hidden layer and the output layer. The sustainable square error is set to 0.2. The training is considered complete when all performance indexes fulfill the criterion, $Q<0.2$.

$$Q^n = \left(Target^n - Output^n\right)^2 \tag{7.10}$$

The modification of the connection follows a gradient-descent method with the delta rule. Each neuron is learning with a different learning rate, α. The variables n and i are representing the output neurons and the connections, respectively.

$$w_{inew}^n = w_{iold}^n - \alpha^n \frac{dQ^n}{dw_i^n} \tag{7.11}$$

In short, the network module and the inference module both receive the belongingness for each set of inputs. In the FNN, the membership values are passed to the six fuzzy rules to determine which value will be used to find the overall inference result, or the centroid, as in the inference module. The ANN calculates the weighted sum of the membership values for each neuron. The networks will adjust the weight of connections under supervised learning. The decision module will generate a signal representing the neuron which will be the greatest output value. The next section will present the simulation results.

7.8 Learning and Testing Results

Two thousand data points were used to train the network. Learning rates are selected after several simulation trails. The learning rates for the neurons are listed in Table 7.2.

	Neuron 1 - R	*Neuron 2 - N*	*Neuron 3 - I*
FNN	0.005	0.045	0.02
ANN	0.00025	0.00006	0.00009

Table 7.2: Learning rates of FNN and ANN.

7.8.1 Result of Learning

The FNN and ANN are trained with a set of training data. There were 2000 input-output pairs of input in the training set. The performance indices were measured during the training process, and the results of every 50th pair of input are plotted in Figures 7.7 to 7.9. The FNN learned to follow the inference results after five iterations with the training set. The learning process of the ANN is slightly longer. It took seven iterations to complete the process. The figures show that the learning process is fairly successful with the predefined learning rates. Especially for the ANN_R neurons and the FNN_N neurons, which converge to a local minimum shortly after being trained once with the training set. The FNN_I neuron started to learn after learning twice. The ANN_N and ANN_I neurons learn with a gradual approach, and converge to their local minima. However, from all the measured performance indices, the ANN_N and ANN_I do not fulfill the criterion. The results indicate a periodical pattern, and this is due to the periodical feature of the training set and repeated training with the same training set.

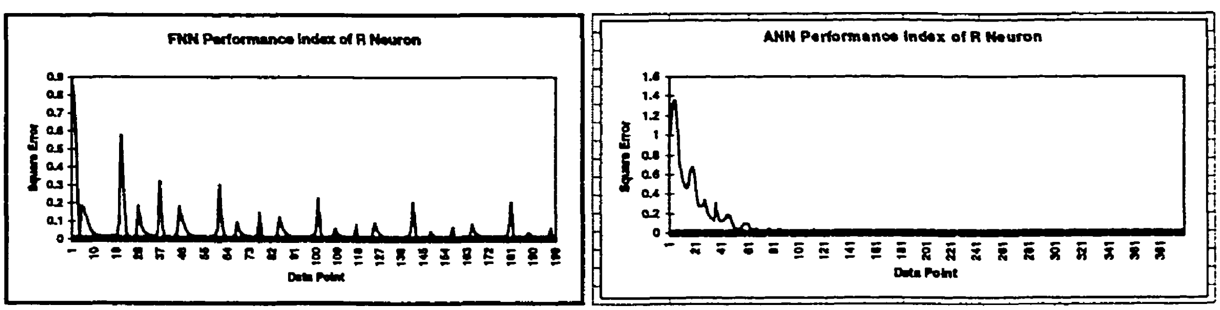


Figure 7.7: Performance indeces, R neurons.

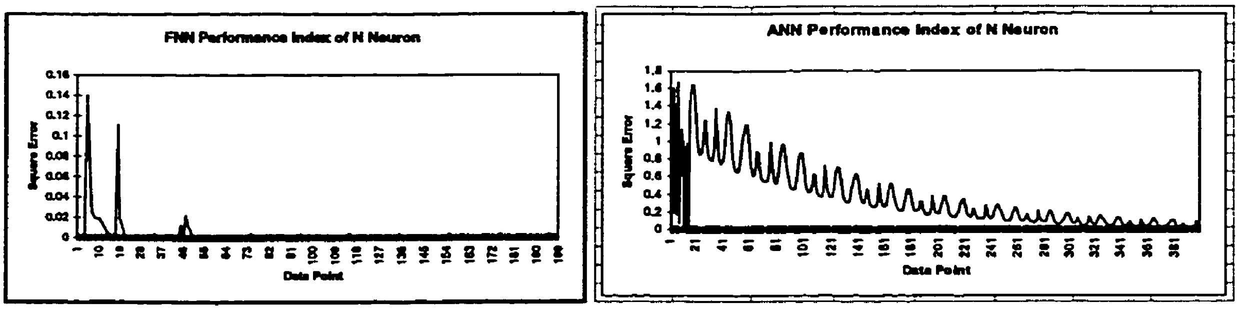


Figure 7.8: Performance indeces, N neurons.

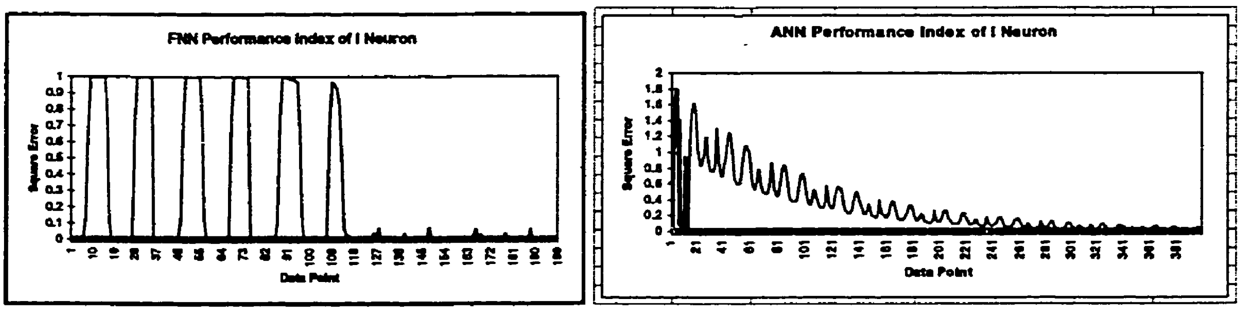


Figure 7.9: Performance indeces, I neurons.

7.8.2 Results of Testing

After the training process, the networks were tested with a set of test data. The cell arrival times for the test set were captured with a GN Nettest InterWatch 95000 protocol analyzer connected to a FORE ASX-200 ATM switch located at TR*Labs*. The buffer size used in the test set is arbitrarily assigned by a random generator. The test set contains 759 pairs of inputs. The simulation results with the test set before and after the training are shown below. The data is recorded for every fifth set of inputs. Figures 7.10 to 7.12 illustrate the signal that the decision-making module would send. As mentioned before, the numbers 1, 2, and 3 correspond to the *R*, *N*, and *I* signals. Before training, the signals generated are very different to the desired ones. After the training, the ANN generated a result identical to the target, and the FNN only made a few mistakes. As shown in Figures 7.10 to 7.12, the graphs are plotted directly on top of the target graph.

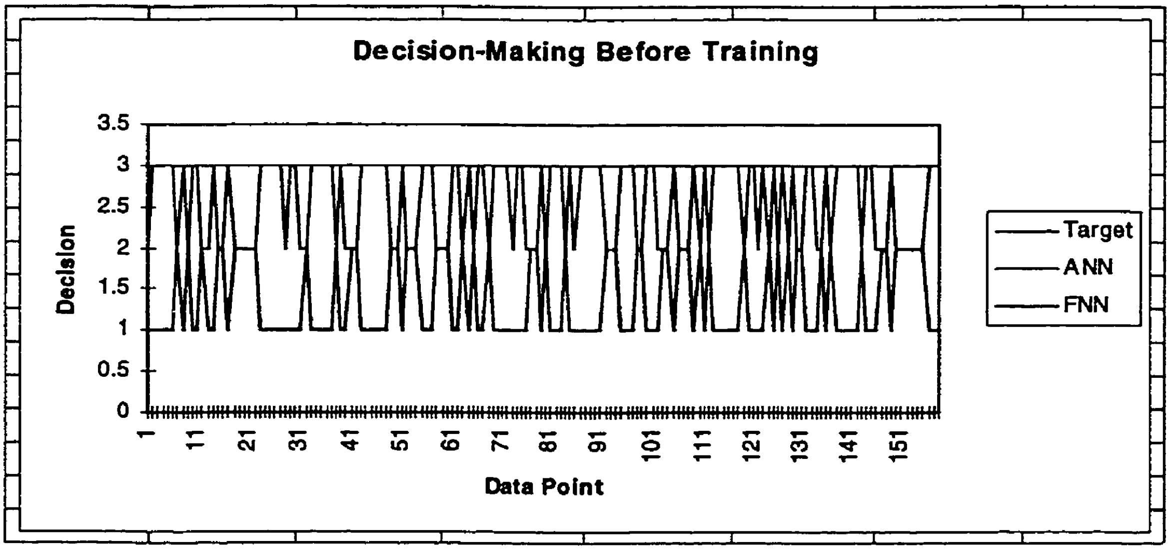


Figure 7.10: Performance before training.

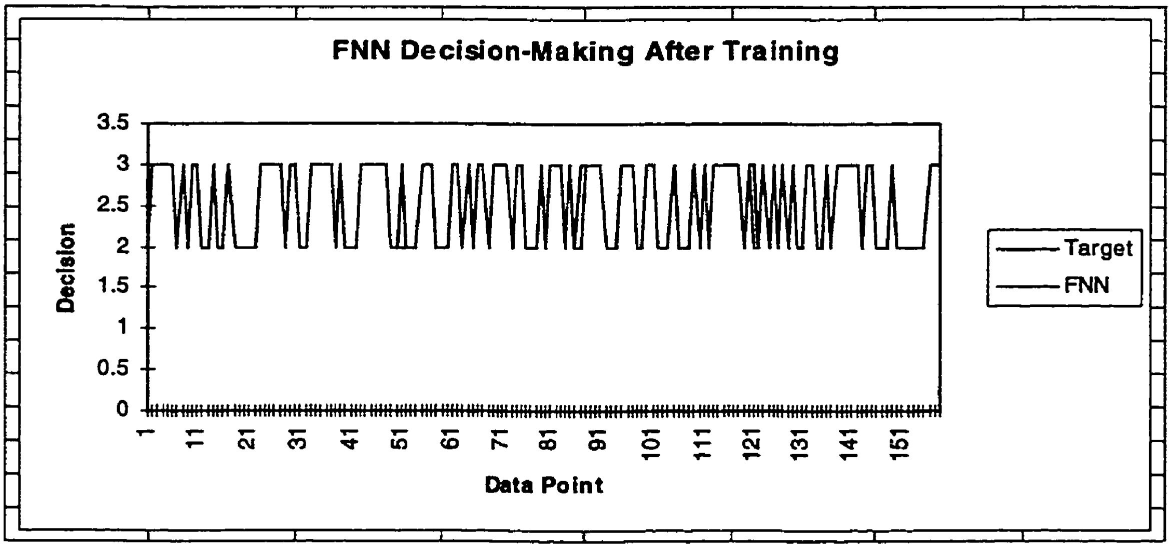


Figure 7.11: Performance of the FNN after the training.

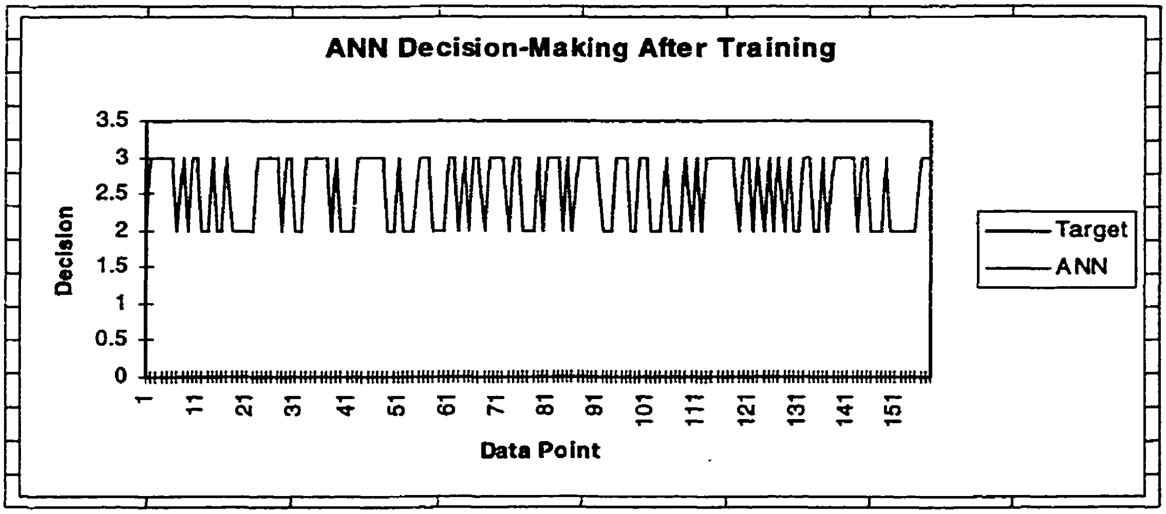


Figure 7.12: Performance of the ANN after the training.

The performance indeces also show positive results. The performance indeces for the FNN and the ANN are improved except for FNN_R. The comparison of the worst situation, the maximum value of the performance index, before and after the training is displayed in Table 7.3. The worst case after the training is obviously improved. ANN_N and ANN_I did not give a satisfactory result, but the overall performance from the decision made by the ANN is superior to that of the FNN. In addition to the simulation result, the computation time for the ANN to carry out the same function is less than half of that of the FNN.

	FNN_R	*FNN_N*	*FNN_I*	*ANN_R*	*ANN_N*	*ANN_I*
Before	0.035	0.848	1.000	1.03	1.02	1.03
After	0.107	0.148	0.0773	0.008	0.452	0.327

Table 7.3: Maximum values of Q.

7.9 Summary and Conclusions

This chapter presented an original ABR control mechanism using fuzzy set theory and logic to maximize the bandwidth utilization of links between ATM switches, and minimize congestion due to traffic overload. An FNN controller and an ANN controller are taught to approximate the fuzzy inference method. The controllers are responsible for deciding if the ABR traffic should be subjected to any changes. The advantage of the mechanism is that it can improve the responsiveness of ATM switches to changes in link traffic. Subsequently, it avoids the unwanted traffic congestion of the link by signaling the previous switch to reduce the injection rate of the ABR traffic.

Testing using the actual network traffic data, shows a very positive result. It is reasonable to say that both controllers make the same decision as the inference method. The FNN learned the inference method faster and the network outputs are closer to the inference values. The ANN did not follow the inference method closely, but the overall decision made by the ANN controller perfectly matches with the decision made by the inference method.

The advantage of the FNN and the ANN over the traditional inference methods is that they require less computational effort, and thus can improve the response of the switch to changes in the traffic condition. The ANN controller also gives the fastest response. This can simplify the fuzzy logic control systems by training an ANN to perform similar tasks. However, this research is still in its preliminary stage. Further research will be required to validate the employability of the new mechanism into the physical switches. Moreover, other kinds of ANN and FNN might provide a better approximation. Using different learning rates may also lead the networks to different local minima, and possibly further enhance performance.

Chapter 8 Conclusions and Recommendations

Traditionally, the behavior of a communication network is determined analytically by going through an intensive mathematical process. With the advent of computer technology, a model can be built on a computer to represent the network. Computer-based modeling can simulate characteristics of network elements, such as the way traffic is generated, how traffic is being served, and the queuing discipline. Network performance can then be determined through simulations. However, just like any other software application, commercial network modeling tools are required to go through verification and validation processes to assure the credibility of the simulation result.

8.1 Summary

The fundamental theories behind network analysis were introduced in this thesis, followed by an introduction to commercial object-oriented network modeling tools (COMNET III and OPNET).

A few network models were built in COMNET III and OPENT. The simulation results provided a base of studying the adequacy of the tools for network behavior studies. The Manitoba Telephone Services (MTS) CCS7 network was then used to illustrate the application of COMNET III for approximating a complex network. In addition a model of the MTS ATM network was built in OPNET to compare the network utilization of Permanent Virtual Connection (PVC) and Switch Virtual Connection (SVC).

The thesis then introduced a flow control mechanism for ATM using an inference method for decision making and its implementation using an Artificial Neural Network (ANN) and Fuzzy Neural Network (FNN). The algorithm can be incorporated into the OPNET ATM model. In what follows the contents of the chapters will be summarized.

Chapter 2 to Chapter 4 illustrated a few test cases performed in both tools. Chapter 2 simulated an M/M/1 model in COMNET III and OPNET. There were some differences in the results due to the inaccuracy of the built-in traffic generators. It is not possible to replace this generation algorithm in COMNET III. The model was designed to run at the sensitive region of the throughput characteristic so that a slight variation in the generated traffic can cause a great difference in throughput. The results showed that both commercial tools have a slightly higher traffic generation rates than the desired rate. The disagreements were clearly visible, since the model was running in the sensitive region. It was observed that a variation of 3% in the traffic rate can cause a 200% variation for OPNET and 500% for COMNET III in the average buffer size and average packet delay. However, if we use the traffic interarrival rate in the analytical formulation that the tools generated, the results agreed perfectly. In other words, the behavior of both models agreed with the analytical result.

In Chapter 3 a second model was implemented to test its behaviour in the domain that is not defined analytically. In this case, the traffic load was greater than the link capacity. The model was an M/D/1 based satellite model. Some interesting and unexpected observations were made in the COMNET III model. The model was intentionally run in the region that the analytical model cannot determine. COMNET III failed to provide the amount of specified traffic load. In contrast, OPNET models provided reasonable simulation results. The specified traffic load was achieved inside and outside the analytical region.

A third model was implemented to validate the library provided by the tools using an ALOHA model. The results were compared to the well known ALOHA throughput characteristic. The COMNET III model utilized the ALOHA link library. The OPNET model utilized the generic OPNET bus topology. The conclusion in Chapter 4 indicates that even though the characteristic of the COMNET III result is similar to the theoretical result, the data points are totally off from the analytical result. Thus it appears that the library was not accurate. The OPNET model which was mostly built from scratch matched the analytical results.

Chapter 5 and Chapter 6 applied the tools to represent physical networks. Chapter 5 approximated the MTS CCS7 network traffic in COMNET III. Chapter 6 studied performance of PVCs and SVCs in an ATM network in OPNET. The results indicated that if the network is providing SVC connections only, then it can support around three times more traffic than when the network provides PVC connections only.

Chapter 7 illustrated a feedback notification-based flow control mechanism, where the preceding switches are told to increase or reduce their ABR traffic rate. The method was based on an inference mechanism that made decisions based on traffic measurements. Two implementations were considered: an FNN and an ANN. Although the FNN learns faster than the ANN, the ANN made better decisions after learning. The ANN also required less computation time than the FNN when making decisions. Therefore, the ANN is a better candidate for implementation of the inference mechanism.

8.2 Conclusions

This thesis focuses on performance and behavior studies of telecommunication networks using two specific commercial object-oriented network modeling tools. The studies concluded that the simulation tools could be used as an alternative for network performance analysis. It was also observed that there could be some problems in representing all kinds of networks. The user will have to verify that the tool is suitable for the desired modeling.

COMNET III can be used to model technologies at the link layer or network layer such as ALOHA and IP. These technologies have to be supported by the library. It is not possible to define new network or link layer technologies. The advantage of COMNET III is that the tool focuses on high level abstraction, since all details are transparent to the users. These libraries include: traffic generators with different probability distributions, different link access mechanisms, and at the network layer there are five kinds of routing mechanisms. The tool is also suited to study new protocols.

Two of the advantages of COMNET III are that it has a friendly user interface and that programming is not required. COMNET III is also a platform independent tool which allows models to be imported among different computers, such as PC, DEC Alpha, and

SUN. In addition, there are no backward compatibility problems. Thus network performance analysts can use the tool to build network models. However, it was observed that the results obtained with this tool can be affected by the traffic generators which failed to provide the required traffic load. The users will have to verify that the parameters of the generator are in the range of operations and the results are reasonable. Regardless of the simplicity of this tool, it is not recommended to rely on the libraries without testing them with a simple model first.

OPNET seemed to be more reliable and more flexible than COMNET III. The problem with OPNET is the complexity of the tool itself. Network analysts would required programming and object-oriented modeling background. Formal training in OPNET would be beneficial, and a programming background would be a pre-requisite. OPNET has some libraries as well. The size of these libraries is continuously increasing – the newer the software version, the bigger the libraries. The libraries simplify the model a little bit. However, not all the important attributes in the libraries are accessible. Often, users are required to modify the libraries in addition to modeling and coding.

Another drawback of OPNET is that the tool is not fully backward compatible. Models built in older versions cannot be accessed through newer versions. An upgrade tool that has a command line interface has been provided, but the converted models may or may not be completely understandable by the newer version. The success of the conversion depends on the complexity of the model. The more complex the model is, the more difficult it is to have a clean conversion. Similarly, not all the data files are forward compatible.

Just like in COMNET III, traffic generators in OPNET have their reliability region. The reliability region in COMNET III appears to be smaller than in OPNET. OPNET does provide the desired traffic load when the required traffic is very light.

8.3 Contributions

This thesis and the work during the completion of it have provided the following contributions:

1. The use of verification methods for network modeling was demonstrated, such as making use of the sensitivity regions of the network characteristics and using the out-of-domain analysis.
2. Different analytical approaches for network analysis were discussed for different kinds of networks.
3. Modeling of the physical and hypothetical networks using the commercial tools, COMNET III and OPNET, were constructed using different levels of abstraction.
4. A traffic modeling study was done for the MTS CCS7 network to match the network into a model.
5. A performance analysis of the MTS ATM network was conducted.
6. A summary of some telecommunication standards, such as SS7 and ATM, was provided.
7. A reliability analysis of the commercial tools was also conducted.
8. A new flow control mechanism for ATM networks was designed and implemented.

8.4 Recommendations

Based on the results of this thesis, the following recommendations are made:

1. Commercial tools can have reliability problems when they are used outside their range of operations, therefore caution would be recommended when using a tool unless it has been validated by a trusted party. Normally, a commercial tool will be tested by the vendor company before it is placed in the market. However, sometimes the known deficiencies are not corrected because there can be complex dependencies between the different computers. Sometimes, there are deficiencies unknown to the vendor themselves. For instance, the libraries could generate errors for a particular network topology and this could be difficult to detect, because there could be many possible network topologies supported by the libraries and it might not be feasible to test all the topologies.

2. It is recommended to perform an analytical study on a given network if possible. Network modeling using a commercial simulation tool does not necessarily provide an easier solution.

3. For complex networks, it is better to study their behavior using a network modeling tool. And yet, tools should be used with caution. The user has to make sure that the parameters are inside the valid range of the tool. If any library will be used, it will be important to make sure that the library has been validated.

4. It is not recommended to use the same tool for all aspects of a study. Some tools, like COMNET III, are tailored for specific applications. Applying tools wrongly could cause a lot of headaches. For example, at the beginning of the performance study of

the MTS network, COMNET III was used, but the tool did not provide ATM components. A lot of time was wasted trying to model it in COMNET III. Eventually, the model was implemented in OPNET.

5. Generic tools, like OPNET, can model almost any network or system. However, it is wise to do some domain analysis to make sure that it is worthwhile to model the network using generic tools. It could easily take ten times longer or more to model a network on a generic tool rather than on a specific tool.

Appendix A Poisson Properties

The material in this appendix is well known, it has been included in this thesis for completeness. All the proofs are derived independently.

Property 1: Poisson Probability Distribution Function

If a process follows an exponential distribution function, then the probability for n number of events to occur in time t can be written as

$$P\left(N(t+s)-N(s)=n\right)=e^{-\lambda t}\frac{(\lambda t)^n}{n!}$$

Comparing to the Poisson Distribution function with rate α,

$$Poi(\alpha)=e^{-\alpha}\frac{\alpha^k}{k!}.$$

The number of events is said to be Poisson Process.

Proof:

It is well known in the probability modeling that the expected value of an exponential distribution function *Exp(*λ*)* is equal to the inverse of its parameter λ. Therefore the mean rate is simply λ. The probability of an event to occur in a small time frame is determined by the rate times the length of the time plus the error term which is an order of the time.

Let $P_n(t)=P(N(t)=n)$, where n indicates the number of events. Therefore the probability for no event occurs in time t is $P_0(t)=P(N(t)=0)$, so that

$$\begin{aligned} P_0(t+h) &= P\{N(t+h)=0\} \\ &= P\{N(t)=0, N(t+h)-N(t)=0\} \\ &= P\{N(t)=0, N(h)=0\} \\ &= P\{N(t)=0\}[1-P\{N(h)\geq 1\}] \\ &= P_0(t)[1-\lambda t+O(h)] \end{aligned}$$

The differential equation of $P_0(t)$, can be found as

$$\begin{aligned} P_0'(t) &= \lim_{h\to 0}\frac{P_0(t+h)-P_0(t)}{h} \\ &= \lim_{h\to 0}\frac{P_0(t)\{1-\lambda h+O(h)\}-P_0(t)}{h} \\ &= \lim_{h\to 0}\left(-\lambda P_0(t)+\frac{O(h)}{h}\right) \\ &= -\lambda P_0(t) \end{aligned}$$

This gives us the Ordinary Differential Equation (ODE)

$$\frac{P_0'(t)}{P_0(t)} = -\lambda$$

Substitute the initial condition $P(N(0)=0)=1$, the probability of initially no event is 1 means $N(0)=0$, into the solution for the ODE. The result is $P_0(t)=e^{-\lambda t}$.

The probability for $n>0$ events to occur in time t,

$$\begin{aligned} P_n(t+h) &= P(N(t+h)=n) \\ &= \sum_{k=0}^{n} P_k(t)P_{n-k}(h) \end{aligned}$$

Recall that $P_1(h)=\lambda h+O(h)$ and $P_0(h)=P(N(h)=0)=1-\lambda h+O(h)$

$$\begin{aligned}P_n(t+h) &= P_n(t)P_0(h)+P_{n-1}(t)P_1(h)+O(h)\\ &= P_n(t+h)(1-\lambda h+O(h))+P_{n-1}(t)\lambda h\\ &= P_n(t)-P_n(t)\lambda h+P_{n-1}(t)\lambda h+O(h)\end{aligned}$$

Rewriting the above equation,

$$\frac{P_n(t+h)-P_n(t)}{h}=-P_n(t)\lambda+P_{n-1}(t)\lambda+\frac{O(h)}{h}$$

As h approaches to zero, the derivative of the equation becomes

$$\frac{dP_n(t)}{dt}=-\lambda P_n(t)+\lambda P_{n-1}(t).$$

Substitutes the initial conditions, the solution becomes

$$P_n(t)=e^{-\lambda t}\frac{(\lambda t)^n}{n!}.$$

This is simply a Poisson distribution function. (QED)

Property 2: Superposition property

The total packet injection rate to the router can be written as $\lambda=\sum_{i=1}^{N}\lambda_i$.

Proof:

Let N=2, when there are only two source nodes, and the system satisfied all the assumptions mentioned above. The processes are called Poisson Processes. The sum of the two sources each with injection rate, $Exp(\lambda_1)$ from node 1 and $Exp(\lambda_2)$ from node 2. The serving rate of the router is the same as in LAN.

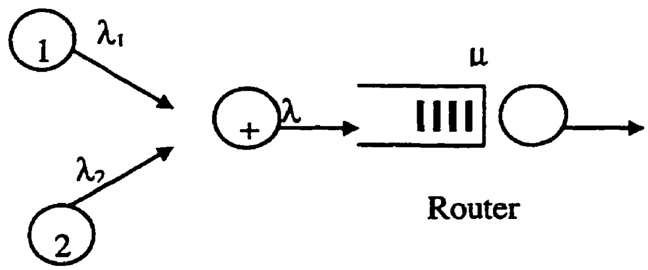


Figure A.1: Two Source nodes with one processor node

The probability of having *n* number of packets at the queue at time *t* is equal to the probability of number of packets sent from node 1 plus the number of packets sent from node2.

$$P(N(t)=n)=P(N_1(t)+N_2(t)=n)$$

If node 1 sent *k* number of packets for *t* = 0, 1, .., n, then the probability of n packets at the queue at time *t* is given by the probability of *k* packets from node 1 with probability of *n-k* packets from node 2. That is

$$P(N(t)=n)=\sum_{k=0}^{n} P(N_1(t)=k, N_2(t)=n-k)$$

Since the processes of sending packets are independent and Poisson, the above equation will become

$$
\begin{aligned}
P(N(t)=n) &= \sum_{k=0}^{n} P(N_1(t)=k)P(N_2(t)=n-k) \\
&= \sum_{k=0}^{n}\left(e^{-\lambda_1 t}\frac{(\lambda_1 t)^k}{k!}\right)\left[e^{-\lambda_2 t}\frac{(\lambda_2 t)^{n-k}}{(n-k)!}\right] \\
&= \frac{1}{n!}e^{-(\lambda_1+\lambda_2)t}\sum_{k=0}^{n}\binom{n}{k}(\lambda_1 t+\lambda_2 t)^n \\
&= e^{-(\lambda_1+\lambda_2)t}\frac{[(\lambda_1+\lambda_2)t]^n}{n!}
\end{aligned}
$$

The derived result is simply a probability of a Poisson process with rate equal $\lambda_1+\lambda_2$. Therefore, for any N number of input sources, the total input rate can be represented by the sum of all the input rates. (QED)

Appendix B ATM Protocol

Asynchronous Transfer Mode (ATM) technology is selected as the link layer for the B-ISDN services. As in the B-ISDN protocol reference model, ATM layer is the layer above the Physical layer (SONET). There is an ATM Adaptation Layer (AAL) sitting on top of the ATM layer. The AAL provides interface for the upper layer applications, such as IP, Frame Relay, Circuit Switching, video, and audio, into the ATM layer.

B.1 B-ISDN Protocol Reference Model

The B-ISDN is logically organized in a layered architecture called the B-ISDN protocol reference model [50], which in turn is organized into three planes, the user plane, the control plane, and the management plane, as in Figure B.1.

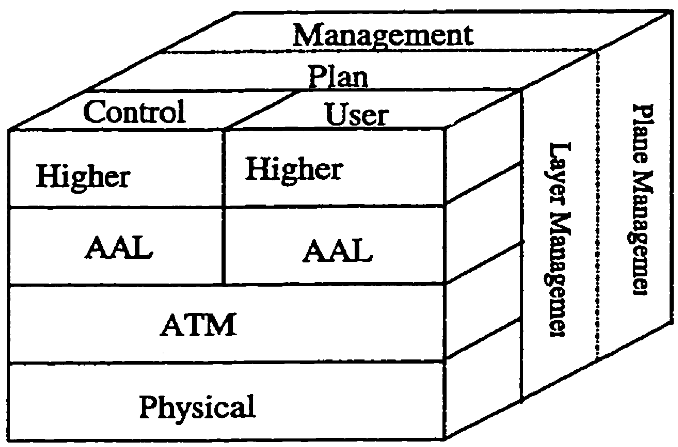


Figure B.1: B-ISDN protocol reference model.

B.1.1 B-ISDN Planes

The user plane deals with the transfer of user information including mechanisms for flow control and error recovery. The control plane is responsible for call and connection control functions, particularly the signaling function that enables the setup, supervision, and release of a call or connection. The management plane includes the layer management and the plane management functions. The layer management has a layered structure and each of its layers handles the specific operations and maintenance (OAM) information flows for the corresponding layers. The plane management is not layered and its task is to provide coordination among all the planes.

B.1.2 B-ISDN Layers

The user plane and the control plane consist of three layers underneath the higher layers. These layers are the ATM adaptation layer, the ATM layer, and the physical layer.

The ATM adaptation layer ensures appropriate service characteristics and divides all data types into 48-octet units that are passed on to the ATM layer. The AAL for the control plane is called signaling AAL (SAAL). The SAAL provides interface for the upper layer signaling such as UNI 3.1 and Q.2931. The SAAL components include the Service Specific Co-ordination Function (SCCF) and Service Specific Connection Oriented Protocol (SCCOP) protocols.

The ATM layer takes the payload sent by the adaptation layer and adds five octets of header information to form a cell. The header information ensures that the cell is sent to the right connection. The physical layer calculates the header error check (HEC) field of the cells, defines the electrical/optical characteristics, and transmits the information to the network.

B.2 Overview of the Architecture of ATM Protocol

An ATM network is a set of ATM switches interconnected by point-to-point ATM links or interfaces. ATM networks are fundamentally connection oriented. Virtual circuits need to be set up across the ATM network prior to any data transfer.

B.2.1 Virtual Path and Virtual Channel Connections

There are two types of ATM connections, namely Virtual Path Connection (VPC) and Virtual Channel Connections (VCC). The VPC connections use the virtual path identifiers (VPI) to distinguish the different connections and the VCC connections use both VPI and virtual channel identifier (VCI) to separate different connections. A virtual path is a bundle of virtual channels. In an ATM network, all VCI and VPI only have local significance across a particular link, and are re-mapped at each ATM switch. Basically, with the known VCI or VPI of an incoming cell and the connection value in a local translation table of the switch, we can determine the outgoing port of the connection and the new VPI/VCI value of the connection on the outgoing link. Then the switches re-transmit the cell on that outgoing link with the appropriate connection identifiers.

B.2.2 ATM Connection Establishment

There are two kinds of connections supported by Asynchronous Transfer Mode (ATM) networks. These connections are called Permanent Virtual Circuit (PVC) and Switched

Virtual Circuit (SVC). This appendix describes how to establish PVC and SVC connections.

PVC connections are all provisioning based. PVC connection setup requires an operator to manually configure the switch ports. When a cell with arrives at one of the input ports, the switch has to know to forward the cell to which output port and replace the cell header with the new VPI/VCI value. Besides specifying the connection at each switch, the operator has to specify the bandwidth allocated to each connection. Once a PVC connection is defined, usually it will exist for weeks, months, or a year, until the customer calls the operator to unsubscribe the service. In that case, the operator will have to disconnect the connections manually as well.

In SVC connection, a connection is established automatically using a set of signaling protocols. ATM is a connection-oriented protocol, the connection must be setup before any data cells can be sent. Connection requests proceed hop-by-hop through the switches of the network to the destination. Switches perform Call Admission Control (CAC) based on traffic descriptor, QoS requirements, and available resources at that switch. If the connection is acceptable, then the request is forwarded on, otherwise "reject" is returned to the caller. If the destination accepts connection, the "accept" is returned, and VPI and VCI is assigned.

B.2.3 Traffic Classes

For either VP or VC with SVC or PVC types of connections, ATM provides four classes of service to setup a connection. Each class is associated with a set of traffic descriptors describing the nature of the call, also known as traffic contract. These classes are defined for different types of higher layer user applications. For example, class A is used for voice applications while class D is for data transfer. These four classes are listed as follows:

- Class A: Continuous Constant Bit Rate (CBR) traffic, such as the circuit emulation
- Class B: Continuous Variable Bit rate (VBR) traffic, such as the voice and video traffic.
- Class C: Connection-oriented Available Bit Rate (ABR) such as user data traffic.
- Class D: Connectionless Available Bit Rate (ABR) such as user data traffic.

B.3 AAL Layer

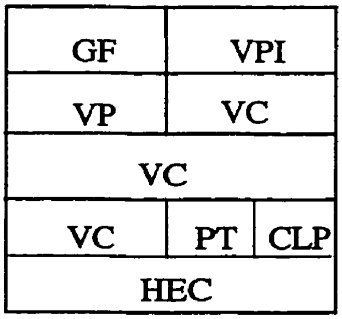
The ATM Adaptation layer (AAL) is defined as enhancing the services provided by the ATM layer to support the functions required by the next higher layer. Different AALs support various protocols to suit the different needs of a range of AAL service users. Recommendation I.362 gives an overview of the basic structure of an AAL and the manner in which it aligns with the service class.

One particular type of AAL service user is the signaling entity wishing to communicate with a peer entity. Any such entity would require functions that are provided above the common part of the AAL specifically designed to facilitate this task.

ATM layer and AAL layer form the B-ISDN third layer. The AAL layer is located on top of the ATM layer. AAL varies with the kind of QoS traffics. The AAL1 is specified for the class A traffic. The traffic classes are listed below. The AAL2 is specified for class B traffic, and AAL3/4 and AAL5 are both for user data traffic.

B.4 ATM Cell Format

ATM cells are 53 bytes long, with five bytes of header and 48 bytes of payload. There are two kinds of ATM cell formats. The interfaces connect ATM end-system to an ATM switch are called User-Network Interfaces (UNI). Another kind of interface is called Network-Node Interfaces (NNI) also known as Network-Network Interfaces that carry out exchange of NNI protocol between two ATM switches. The figures below show the header formats of the UNI. The only difference between UNI and NNI format is that the NNI header does not have the Generic Flow Control (GFC) field. Instead NNI has a long Virtual Path Identifier (VPI) field to hold the longer address in the WAN environment.



GFC	Generic Flow Control
VCI	Virtual Channel Identifier
VPI	Virtual Path Identifier
PT	Payload Type
CLP	Cell loss Priority
HEC	Header Error Control

Figure B.2: UNI format of the cell header.

- The GFC is absent in the NNI header, because it is envisaged to provide contention resolution, and simple flow control for shared medium-access and arrangements at the customer premises equipment (CPE). NNI eliminated GFC to provide a longer VPI field.

- The PT is a three-bit field defining the types of information carried by the cells, including operation, administration and maintenance (OAM) information or user information.

- The CLP is a one-bit field indicating the priority of the cell. By default the bit is defined as 0. If the bit is set to 1 that means the cell has lower priority. The cells with lower priority might be discarded by the network, depending on the network condition.

- HEC helps to reduce the cell loss and misrouting due to cell header errors. It performs Cyclic Redundant Check (CRC) calculation on the first four bytes of the header field. However, as mentioned before, ATM does not have error control on the actual payload or data itself.

B.5 Congestion Control

- Before setting up a connection, the end-users must negotiate the traffic contract. The traffic contract includes QoS and the connection traffic descriptor. The well known Peak Cell Rate (PCR), Cell Drop Variation (CDV), and the conformance definition. The conformance definition may be based on the Genetic Cell Rate Algorithm (GCRA). The algorithm is a leaky bucket algorithm, and can be described as a function of rate and length. The parameters are related to the PCR and CDV respectively.

- Usage Parameter Control (UPC) monitors the VPC cells pass the VP switches and VCC cells pass the VC switches. During heavy traffic conditions, UPC will drop the excess cells and change the CLP to lower priority when it is necessary. It also monitors the PCR of the cells to ensure that the users do not exceed the negotiated rate.

- Other congestion control mechanisms are Network Resource Management (NRM), Selective Cell Discarding (SCD), Connection Admission Control, and traffic shaping. The functions of NRM are grouping the related VCC into VPC to simplify the traffic management and resource allocation. SCD is a supplement to UPC. It is responsible for dropping the cell with CLP=1. CAC is used during the call setup. It only accepts a new connection when there is sufficient resources for the required QoS of the traffic. When a new connection is accepted, CAC passes the traffic parameters to UPC and allocate the routes.

B.5.1 Call Admission Control

The purpose of an admission control algorithm is to decide, at the time of call arrival, whether or not a new call should be admitted into the network. A new call is admitted if and only if its QoS constraints can be satisfied without jeopardizing the QoS constraints of existing calls in the network. CAC also determines the parameters of the connection that need to be passed to UPC and also allocates and routes network resources for the connection.

Admission Control decision is made based on the traffic contract which is composed of connection traffic descriptors, QoS requirements, and conformance definitions (may be based on GCRA and enforced by UPC).

1. Connection Traffic Descriptor: Peak Cell Rate (PCR), Sustained Cell Rate (SCR), Cell Delay Variation Tolerance (CDVT) and Maximum Burst Size (MBS).
2. QoS Requirements: tolerable cell loss, cell delay, and delay variation.

Beside the CAC, there is a Network Resource Management (NRM) at the switch. The NRM groups the related VCCs into same VPC to simplify traffic management and resource allocation.

B.5.2 During a Call

When a call is established, there is a possibility that a call can exceed the negotiated contract during CAC. In that case, it is necessary to carry out the Generic Cell Rate Algorithm and the Usage Parameter Control (UPC) or Policy Function.

The GCRA checks the conformance of the ATM cells arriving at a UNI. One method of implementing the GCRA is as a continuous-state of Leaky Bucket Algorithm.

The UPC consists of a set of actions so as to protect the network resources and maintain a connection's QoS. UPC must monitor the PCR to ensure that the user never exceeds the traffic contract for a connection. The conforming cells will be passed to the output port, and the non-conforming cells will be tagged or discarded.

B.5.3 Other Congestion Controls

1. Selective cell discarding (SCD): This function discards some cells with CLP=1, and it is used to augment the capabilities of UPC.

2. Traffic Shaping: This function is to mold the non-conforming streams of cells to conform to the traffic contract.

Appendix C List of Abbreviations

ATM	Asynchronous Transfer Mode
AAL 5	ATM Adaptation Layer type 5
ABR	Available Bit Rate
ACM	Address Complete Message
AE	Application Entity
ANM	Answer Message
ANN	Artificial Neural Network
ANSI	American National Standard Institute
ASE	Application Service Element
BISDN	Broadband Integrated Services Digital Networks
bps	Bits per second
BRI	Basic Rate Interface
CAC	Call Admission Control
CAN	Campus Area Network
CBR	Constant Bit Rate
CCITT	Consultative Committee on International Telegraph and Telephone
CCS6	CCITT Signaling System No. 6
CCS7	Common Channel System 7
CDV	Cell Delay Variation
CLP	Cell Loss Priority
CO	Central Office

CPE	Customer Premises Equipment
CRC	Cyclic Redundant Check
CSMA	Carrier Sense Multiple Access
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DAP	Data User Part
DQDB	Distributed Queue Dual Bus
FDDI	Fiber Distributed Data Interface
FNN	Fuzzy Neural Network
GAN	Global Area Network
GCRA	Generic Cell Rate Algorithm
GFC	Generic Flow Control
GUI	Graphical User Interface
HEC	Header Error Check
IAM	Initial Address Message
IDC	Index of Dispersion for Counts
IN	Intelligent Network
IP	Internet Protocol
IP	Intelligent Peripheral
ISDN	Integrated Services Digital Networks
ISP	Intermediate Service Part
ISUP	ISDN User Part
ITU	International Telecommunication Union
IWU	Internetworking Unit

LAN	Local Area Network
MAC	Medium Access Control
MAN	Metropolitan Area Network
MAU	Multiple Access Unit
MBS	Maximum Burst Size
MTP	Message Transfer Part
MTS	Manitoba Telecom Services
NNI	Network-Node Interfaces
NRM	Network Management Resource
NSP	Network Service Part
OAM	Operation and Maintenance
OMAP	Operations Maintenance and Administration Part
OOA	Object-Oriented Analysis
OOM	Object-Oriented Modeling
OSI	Open Systems Interconnection
OSIRM	OSI Reference Model
PCR	Peak Cell Rate
PCS	Personal Communication Systems
PDU	Protocol Data Unit
PF	Policing Function
PRI	Primary Rate Interface
PS	Priority Shaping
PSTN	Public Switching Telephone Networks

PT	Payload Type
PVC	Permanent Virtual Circuit/Connection
QoS	Quality of Service
REL	Release Message
RLC	Release Complete
SAAL	Signaling AAL
SCCF	Service Specific Co-ordination Function
SCCOP	Service Specific Connection Oriented Protocol
SCCP	Signaling Connection Control Part
SCD	Selective Cell Discarding
SCP	Service Control Point
SCR	Sustained Cell Rate
SL	Signaling Link
SMDS	Switched Multi-megabit Data Service
SMS	Service Management System
SONET	Synchronous Optical Network
SP	Signaling Point
SSP	Service Switching Point
STD	State Transition Diagrams
STP	Signal Transfer Point
SUS	Suspend Message
SVC	Switched Virtual Circuit/Connection
TC	Transaction Capability

TCAP	Transaction Capabilities Application
TCAP	Transaction Capabilities Application Part
TCP	Transaction Control Protocol
TUP	Telephone User Part
UNI	User-Network Interfaces
UPC	Usage Parameter Control
VBR	Variable Bit Rate
VCC	Virtual Channel Connection
VCI	Virtual Circuit Identifier
VPC	Virtual Path Connection
VPI	Virtual Path Identifier
WAN	Wide Area Network

Appendix D References

[1] N. Prabhu and U. Bhat, "Some First Passage Problems and their application to queues," *Sankhya A*, pp281-292, 1963.

[2] A. Alles, *ATM Internetworking*. Las Vegas: Engineering InterOp, March 1995.

[3] D. McDysan, and D. Spohn, *ATM Theory and Application*. New York: McGraw Hill, 1994.

[4] W. Stallings, *Data and Computer Communications 3rd Ed.* New York: MacMillan, 1991.

[5] P. Knepell, and D. Arangno, *Simulation Validation*. Los Alamitos, CA: IEEE Computer Society Press, 1993.

[6] C. Hunt, *TCP/IP Network Administration*. California: O'Reilly, 1994.

[7] S. Papat, *Object-Oriented Network*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1994.

[8] B. Kumar, *Broadband Communications*. New York: McGraw Hill, 1994.

[9] J. Webster, "ISDN Update," *Telecommunications*, v. 29, no. 9, pp101-102, Sep 1995.

[10] R. Azarmsa, *Telecommunications: A Handbook for Educators*. New York: Garland Publishing, 1993.

[11] G. McClain, *Handbook of Networking & Connectivity*. Boston: AP Professional, 1994.

[12] B. Linberg, *Digital Broadband Networks & Services*. New York: McGraw-Hill, 1994.

[13] A. Law, and W. Kelton, *Reference on the Statistical Treatment of Simulation Experiments, 2nd Ed.* New York: McGraw-Hill, 1991.

[14] H. Neston, *Newton's Telecom Dictionary 5th Ed.* New York: Telecom Library Inc., 1992.

[15] J. Powers Jr., and H. Stair II, *Megabit Data Communications*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1990.

[16]J. Green, *The Dow Jones-Irwin Handbook of Telecommunications*. Illinois: Dow Jones-Irwin, 1986.

[17] W. Stallings, *ISDN and Broadband ISDN with Frame Relay and ATM, 3rd Ed.* Englewood Cliffs, NJ: Prentice-Hall Inc., 1995.

[18] S. McGovern, *Digital, Analog, and Data Communication 2nd Ed.* Englewood Cliffs, NJ: Prentice-Hall Inc., 1986.

[19] D. Honig, and K. Hoover, *Desktop Communications*. New York: John Wiley & Sons Inc., 1990.

[20] T. Bartee, *Data Communications, Network, and Systems, 2nd Ed.* USA: SAMS, 1992.

[21] Y Bourlas and A. Kwok, "Modeling and Simulation of the Common Channel Signaling (CCS7) Network," *IEEE WESCANEX 1995*, vol. 2, pp42-46, 1995.

[22] P. Verma, *ISDN System: Architecture, Technology, and Applications*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1990.

[23] J. Griffiths, *ISDN Explained: Worldwide Network and Applications Technology 2nd Ed.* Britian: John Wiley & Sons, 1992.

[24] H. Helgert, *Integrated Services Digital Networks: Architectures, Protocols, Standards.* Reading, MA: Addison-Welsey,

[25] P. Verma, *ISDN Systems: Architecture, Technology, and Applications*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1990.

[26] F. Burg and N. Iorio, "Networking of Networks: Internetworking According to OSI," *IEEE Journal on Selected Areas in Communications*, pp1131-1142, Sep 1989.

[27] P. Kuhn, C. Pack, and R Skoog, "Common Channel Signaling Network: Past, Present, Future," *IEEE Journal on Selected Areas in Communications*, pp383-394, April 1994.

[28] D. Manfield, G. Millsteed, and M. Zukerman, "Performance Analysis of SS7 congestion Controls Under Sustained Overload," *IEEE Journal on Selected Areas in Communications*, pp405-414, April 1994.

[29] M. Rumsewicz, "On the Efficacy of Using the Transfer Controlled Procedure During Periods of STP Processor Overload in SS7 Networks," *IEEE Journal on Selected Areas in Communications*, pp415-423. April 1994.

[30] D. Smith, "Effects of Feedback Delay on the Performance of the Transfer-Controlled Procedure in Controlling CCS Network Overloads," *IEEE Journals on Selected Areas in Communications*, pp424-432, April 1994.

[31] V. Bolotin, "Modeling Call Holding Time Distributions for CCS Network Design and Performance Analysis," *IEEE Journal on Selected Areas in Communications*, pp433-438, April 1994.

[32] D. Duffy, A. McIntosh, M. Rosenstein, and W. Willinger, "Statistical Analysis of CCSN/SS7 Traffic Data from Working CCS Subnetworking," *IEEE Journal on Selected Areas in Communications*, pp544-551.

[33] C. Sunshine, "Network Interconnection and Gateways," *IEEE Journal on Selected Areas in Communications*, pp4-11, Jan 1990.

[34] M. Woodward, *Communication and Computer Networks*. Los Alamitos, CA: IEEE Computer Society Press, 1994.

[35] ATM Forum Traffic Management v. 4.0.

[36] C. Douligeris and G. Develokos, "A Fuzzy Logic Approach to Congestion Control in ATM Networks," *International Conference on Communications' 1995*, pp1969-1973.

[37] Y. Park, V. Cherkassky, and G. Lee, "Omega Network-Based ATM Switch with Neural Network-Controlled Bypass Queueing and Multiplexing", *IEEE Journal on Selected Areas in Communications*, pp1471-1480, Dec 1994.

[38] E. Yu, and C. Chen, "Traffic Prediction Using Neural Networks", in *International Conference on Communications' 1993*, pp991-995.

[39] T. Ndousse, "Fuzzy Neural Control of Voice cells in ATM Networks", *IEEE Journal on Selected Areas in Communications*, pp1488-1494, Dec 1994.

[40] A. Tarraf, I. Habib, and T. Saadawi, "Characterization of Packetized Voice Traffic in ATM Networks using Neural Networks", *IEEE International Conference on Communications*, pp996-1000, 1993.

[41] A. Bonde, Jr. and S. Ghosh, "A Comparative Study of Fuzzy Versus "Fixed" Thresholds for Robust Queue Management in Cell-Switching Networks," *IEEE/ACM Transaction on Networking*, pp337-344, Aug 1994.

[42] Y. Chang, N. Golmie, and D. Su, "A Rate Based Flow Control Switch Design for ABR Service in an ATM Network," http://www. nist.org/

[43] A. Kwok and R. McLeod, "ATM Congestion Control Using a Fuzzy Neural network," *IEEE Canadian Conference on Electrical and Computer Engineering 96*, Calgary, 1996.

[44] ANSI T1.111.1 Functional Description of the Message Transfer Part, 1991.

[45] ANSI T1.111.2 Signalling Data Link, 1991.

[46] ANSI T1.111.3 Signalling Link, 1991.

[47] ANSI T1.111.4 Signalling Network Functions and Messages, 1991.

[48] ANSI T1.111.5 Signalling Network Structure, 1991.

[49] M. Schwartz, *Telecommunication Networks, Modeling and Analysis.* California: Addison Wesley, 1987.

[50] ATM Forum User Network Interface v.3.1, 1994.

MODELING, SIMULATION, AND PERFORMANCE EVALUATION OF TELECOMMUNICATION NETWORKS

M.Sc. Thesis Presentation

by Alice Kwok

General Overview

- Introduction to network modeling
- Modeling methods and Techniques
- Modeling tools
  - COMNET III and OPNET
- Basic network models
  - Simple model
  - Satellite network model
  - Aloha network
- CCS7 network modeling
- ATM network modeling
- Congestion control mechanism for ABR traffic

Introduction to Network Modeling

- **Network model is a representation of a network under study**
- **Aspects:**
  - physical, logical, and functional
- **Goal:**
  - to manipulate the parameters and to observe the behaviour in the different circumstances.
- **How? (By using OOM techniques)**
  - Abstraction: extract important features within the abstraction barrier
  - Encapsulation: identifying the internal implementation details and present it as a whole
  - inheritance: classification process to categorize the objects with similar properties into an object class

Modeling Methods and Techniques

- **Empirical Model - physical network**
- **Analytical/Theoretical Model - queuing theory**
  - arrival process
  - service process
  - number of servers
  - waiting room
  - customer population
  - service discipline
- **Simulation Model - computer based modeling**
  - COMNET III
  - OPNET

Modeling Tools - COMNET III & OPNET

COMNET III

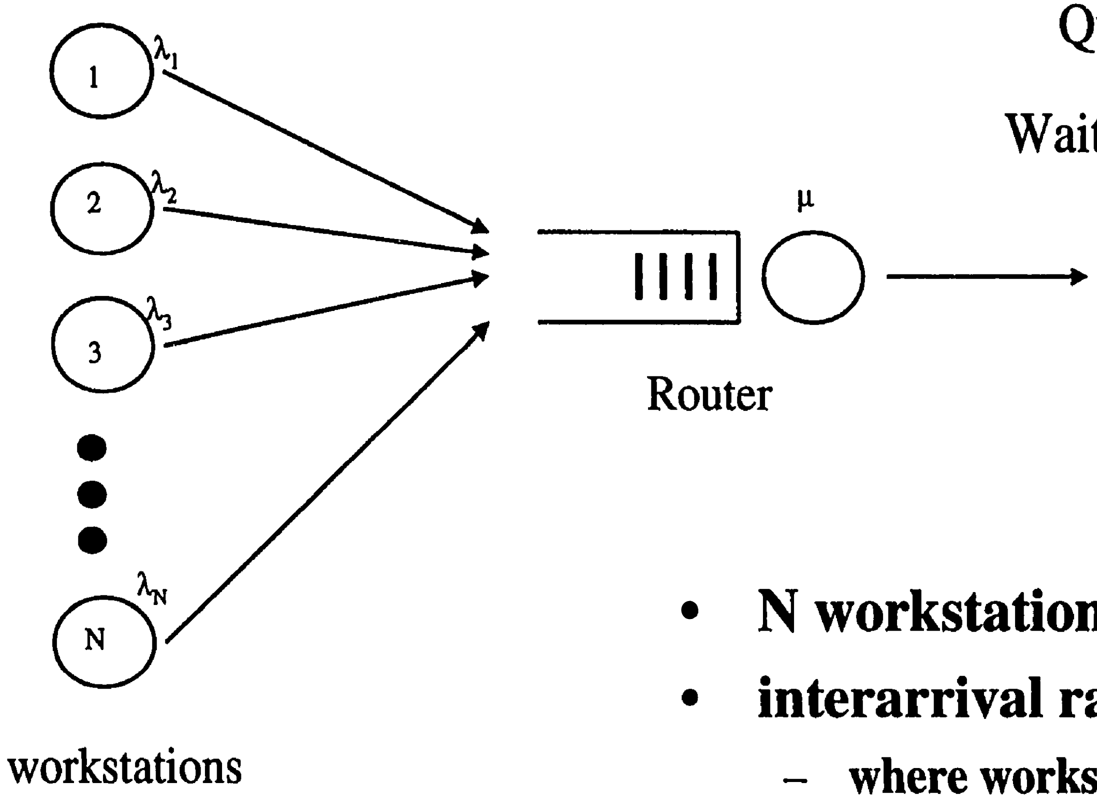
- More user friendly
- High level of abstraction
- Hierarchical approach
- Drag and drop network components
  - clouds, subnetworks, network topologies, nodes and links.
- All network configurations are done through the GUI
- No programming required
- All-in-one tool
- Does not support mobile but geo-stationary networks
- Portable models for different platforms

OPNET

- Lower level of abstraction
- object representation provides better flexibility
- Hierarchical approach
- Programming language
  - OPNET library
  - C programming language
- A set of eight software tools
- Support mobile networks
- Runs on SUN workstations

Comparison on the tools using Simple Network Models

Simple Network - M/M/1 Markov Chain

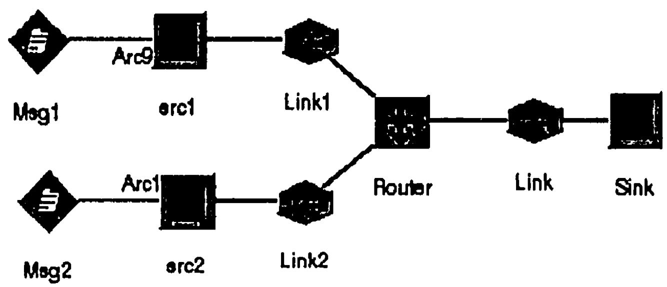


Queue size = L = $\frac{\lambda^2}{\mu(\mu-\lambda)}$

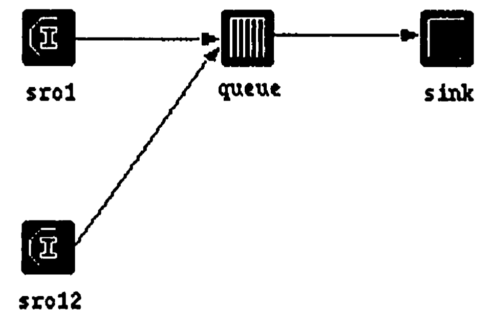
Waiting time = W = $\frac{\lambda}{\mu(\mu-\lambda)}$

- **N workstations and a router (N=2)**
- **interarrival rate = $\lambda = \sum \lambda_I$=7*8000bps**
  - **where workstation i interarrival rate = λ_I**
- **router service rate = μ=58000 bps**

Simple Network - Results



COMNET III



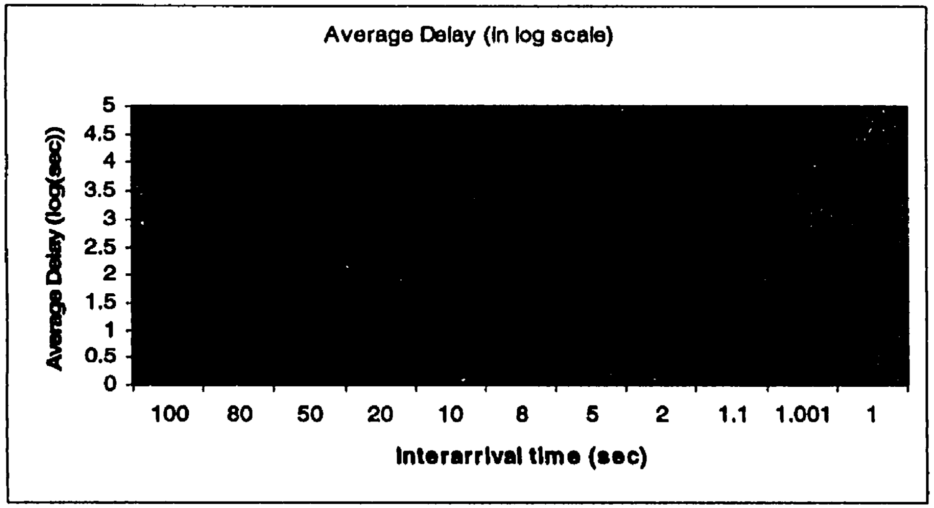
OPNET

	Analytical Calculation	COMNET III Model	OPNET Model
Average packets (src1/Msg1)	5	5	5
Average packets (src2/Msg2)	2	2	2
Average packet delay time (sec)	3.86	19.6	7.[illegible]
Maximum delay time (sec)	N/A	50.1	12.5
Average buffer size (KB)	27	137.5	50
Maximum buffer size (KB)	N/A	347	90

Higher interarrival rates

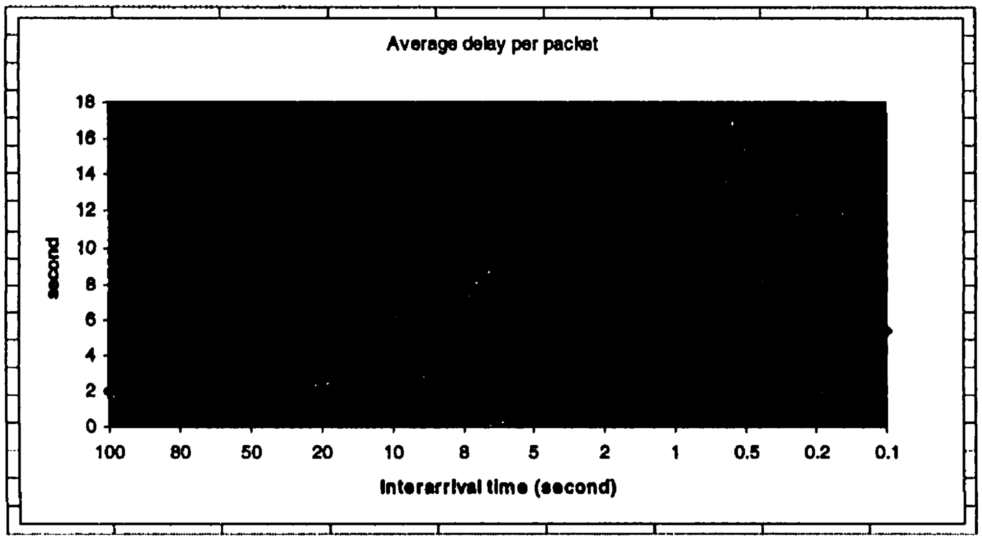
Interarrival Time (bps)	56000	57000	57500	57600
Average Buffer Size (pkts)	27	56	114	143
Average packet delay time (sec)	3.86	7.86	15.86	19.8
Similar Model	Analytical	OPNET	N/A	COMNET III

Satellite Network - M/D/1

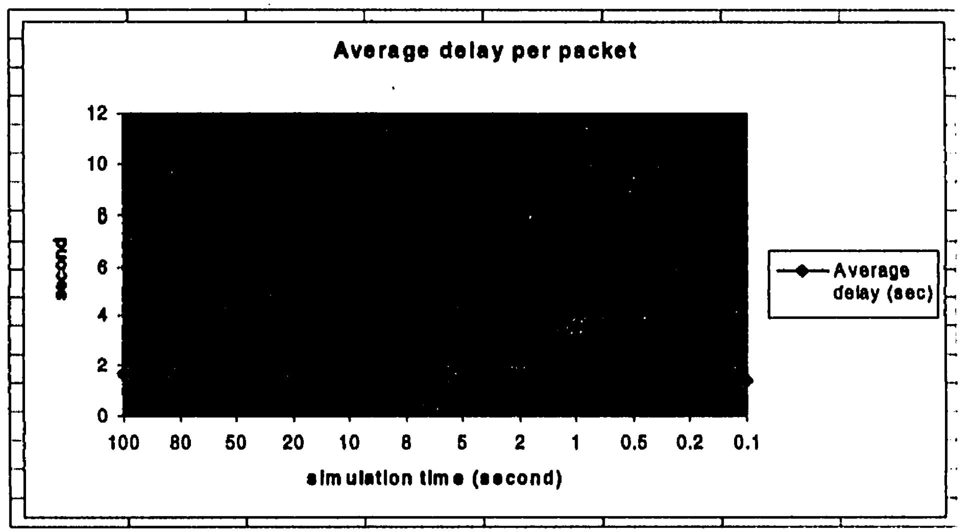


Analytical model

COMNET cannot generate enough traffic beyond the analytical range

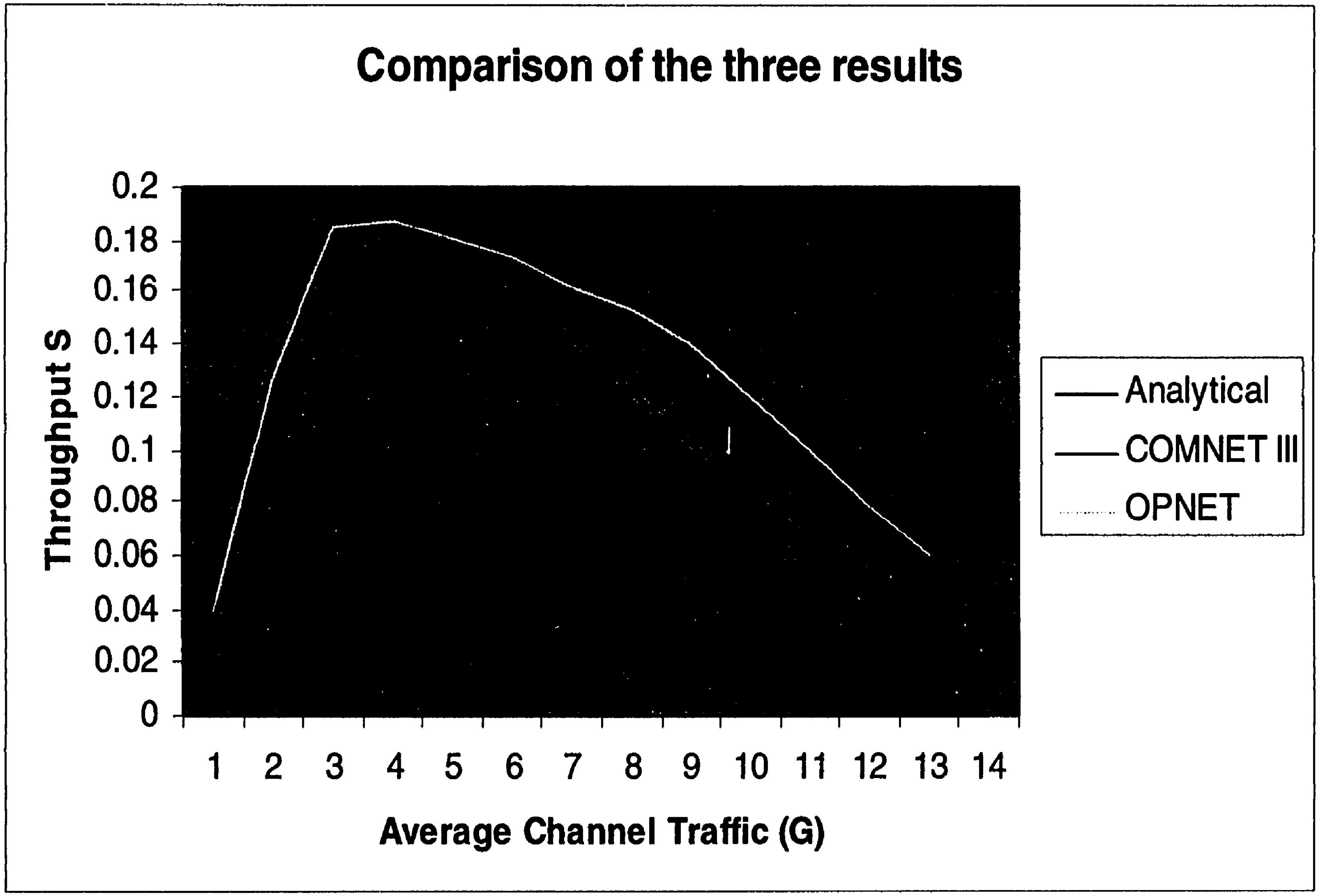


OPNET model



COMNET III model

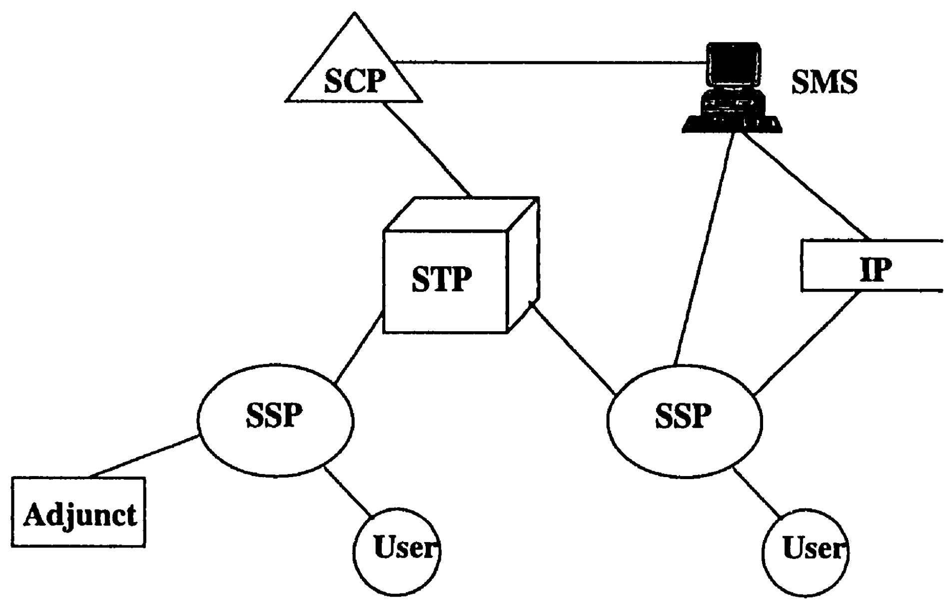
Aloha Network



CCS7 Network Simulation:

Approximate the SS7 Signaling traffic

CCS7 Network - IN architecture

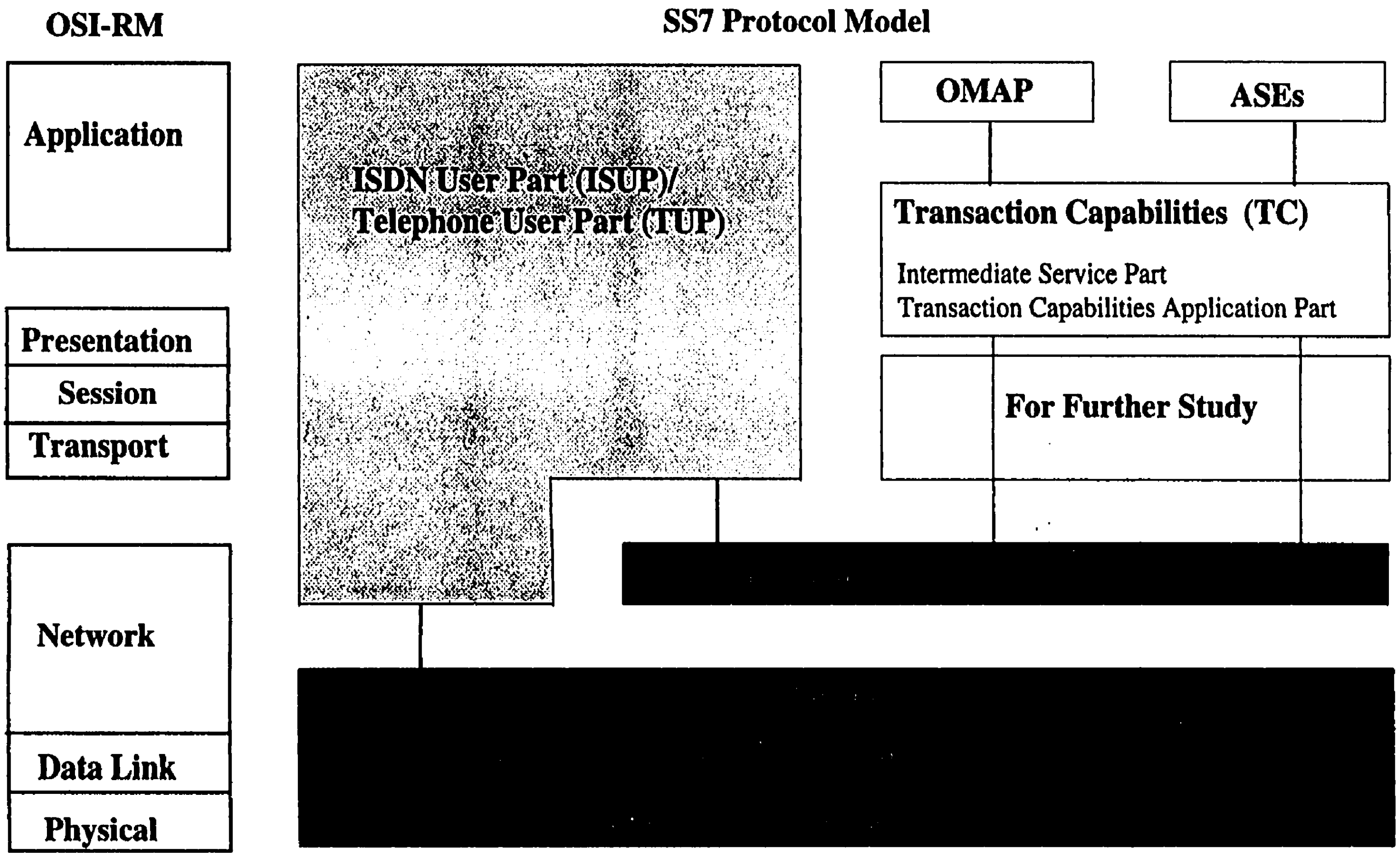


Services:
- call setup
- call termination
- network management

Functions:
- supervising
- alerting
- addressing

SMS - service management system
SCP - service control point
STP - signal transfer point
SSP - service switching point
IP - intelligent peripheral

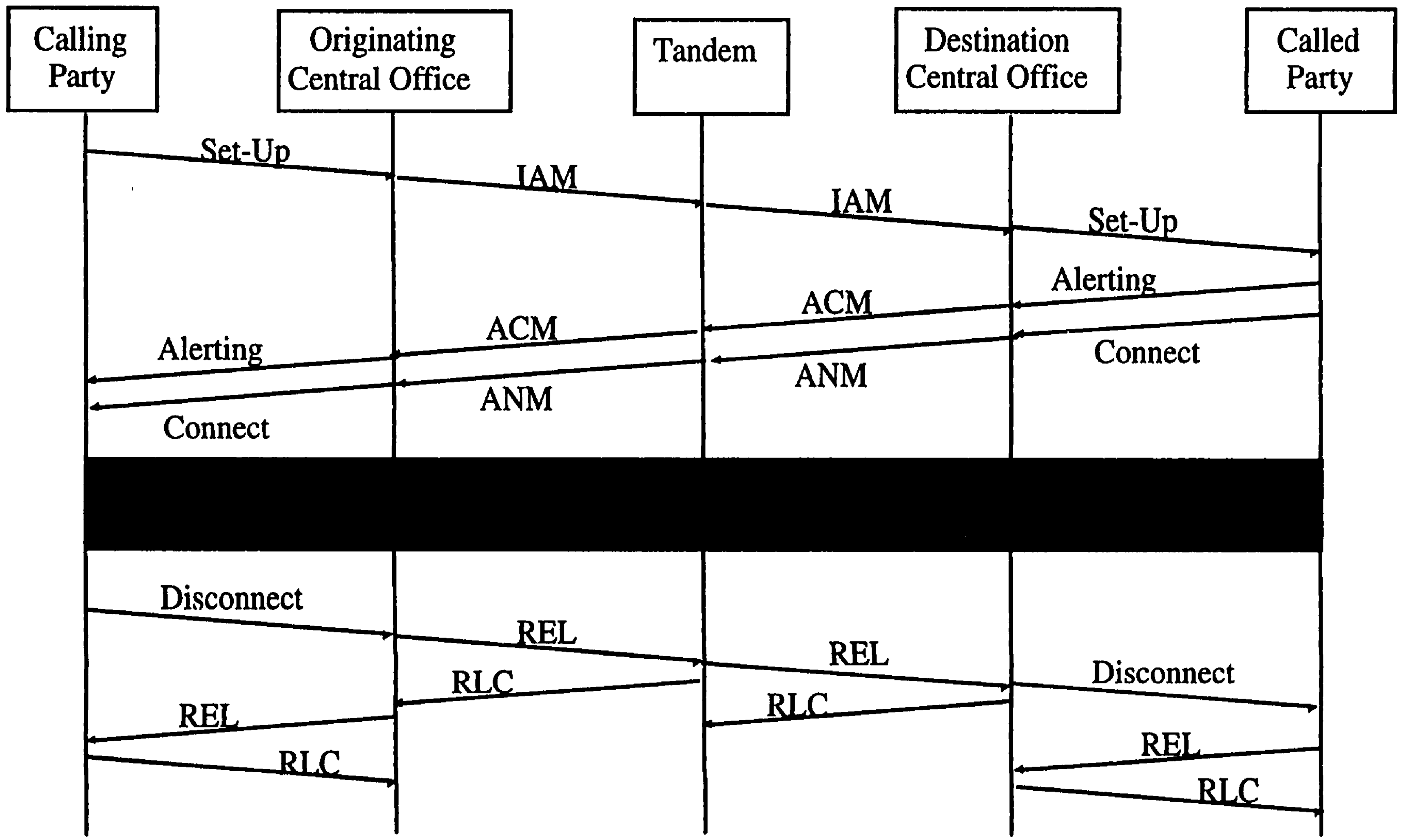
CCS7 Protocol Architecture



OMAP = Operations Maintenance and Administration Part
ASE = Application Service Element

Network Service Part

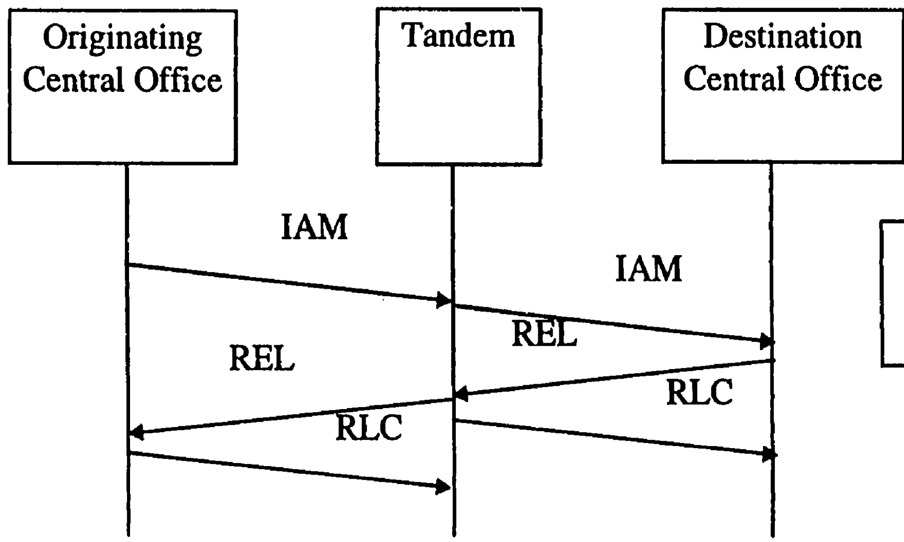
Local Call Setup and Termination



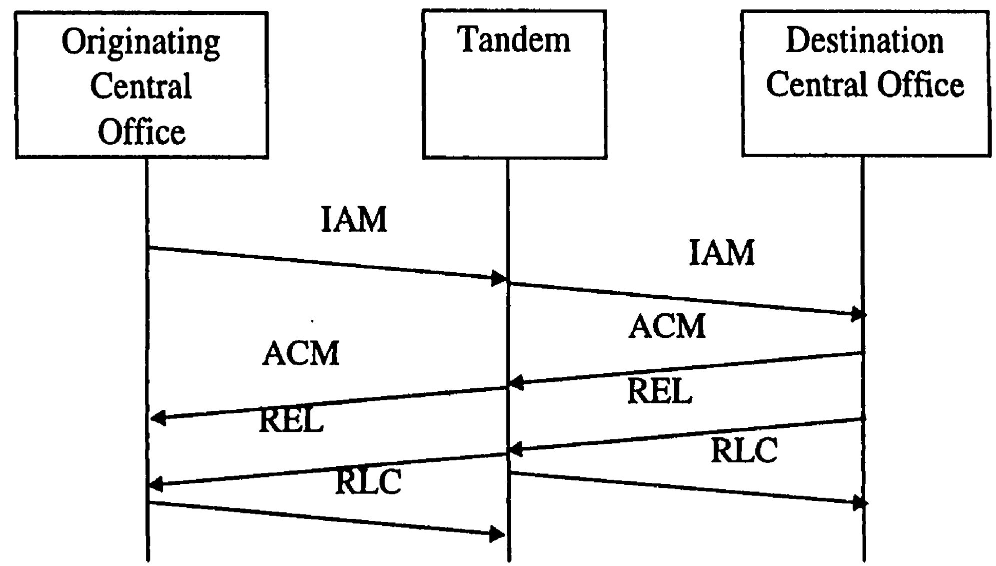
IAM = initial address message
ACM = address complete message
ANM = answer message

REL = release message
RLC = release complete

Local Call Busy and Not Answered

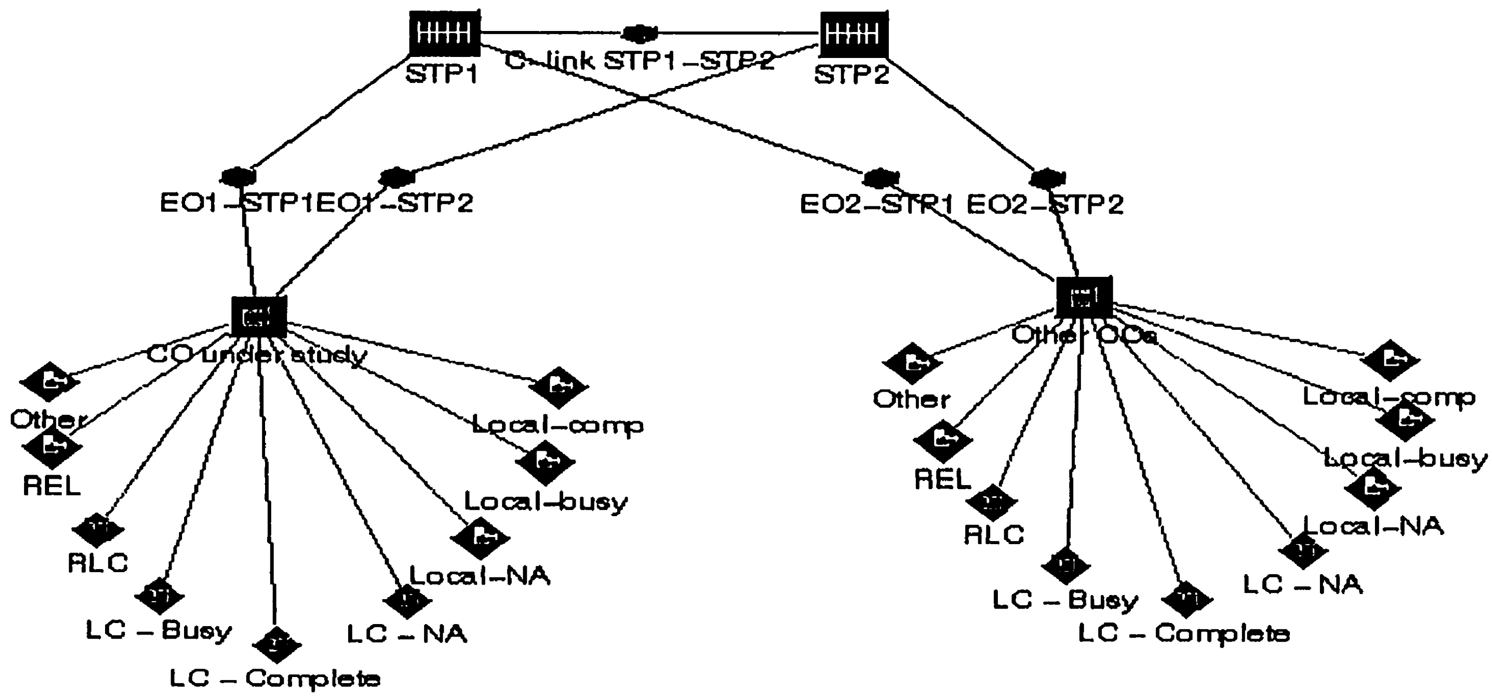


Call Busy



Call Not Answered

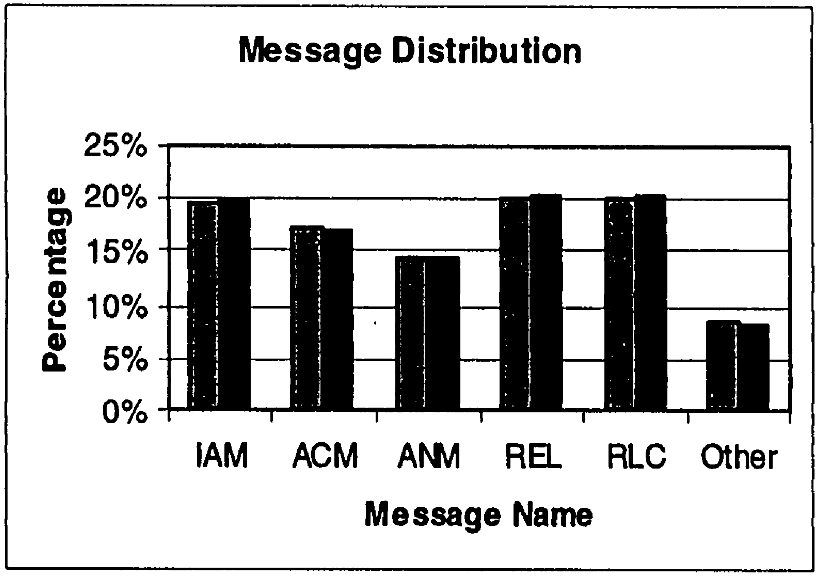
MTS Local Call Network Model on COMNET III

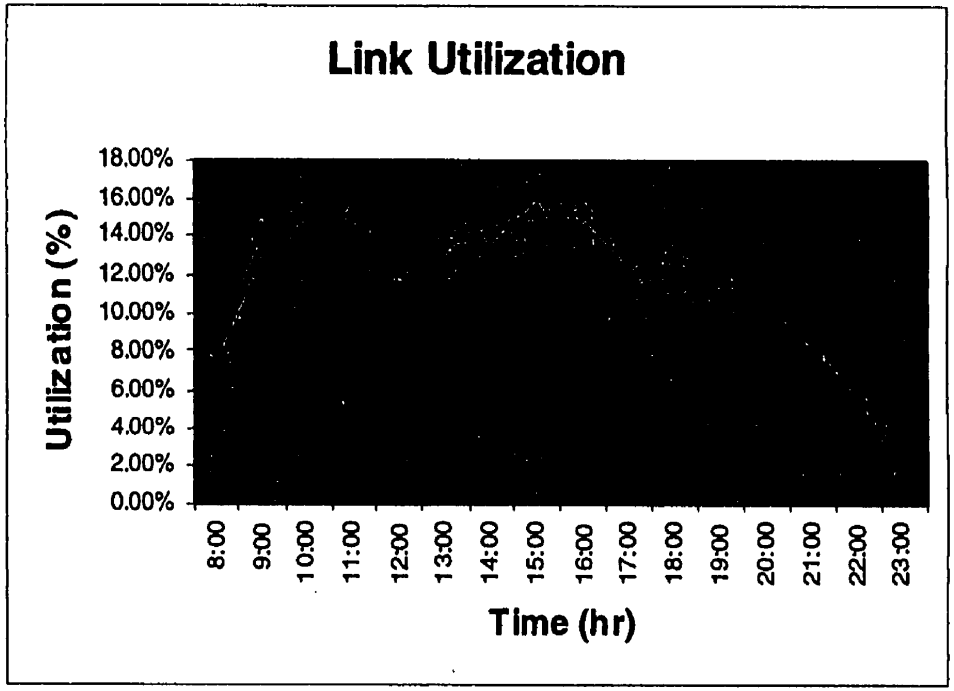


Call Type	Message Originating Node	Message Response Node	Percentage of the traffic load
Call Complete	Local-comp	LC-Complete	75%
Call Busy	Local-Busy	LC-Busy	12.5%
Call Not Answered	Local-N/A	LC-N/A	12.5%

IAM = (27-57, mean 52) octets
ACM = 11 octets
ANM = 9 octets
REL = 12 octets
RLC = 8 octets
Other = 10 octets ~8% total traffic

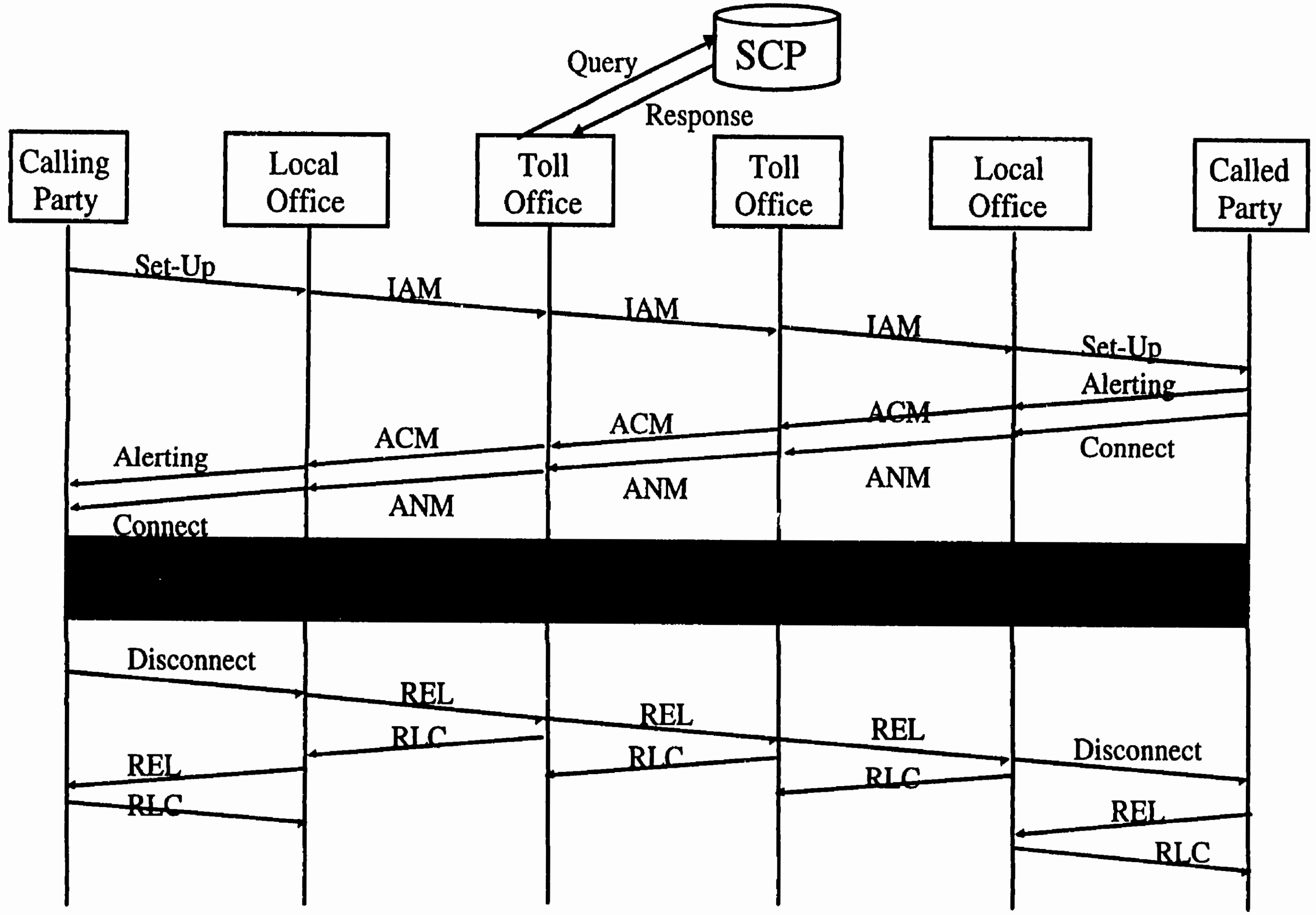
Approximation Results



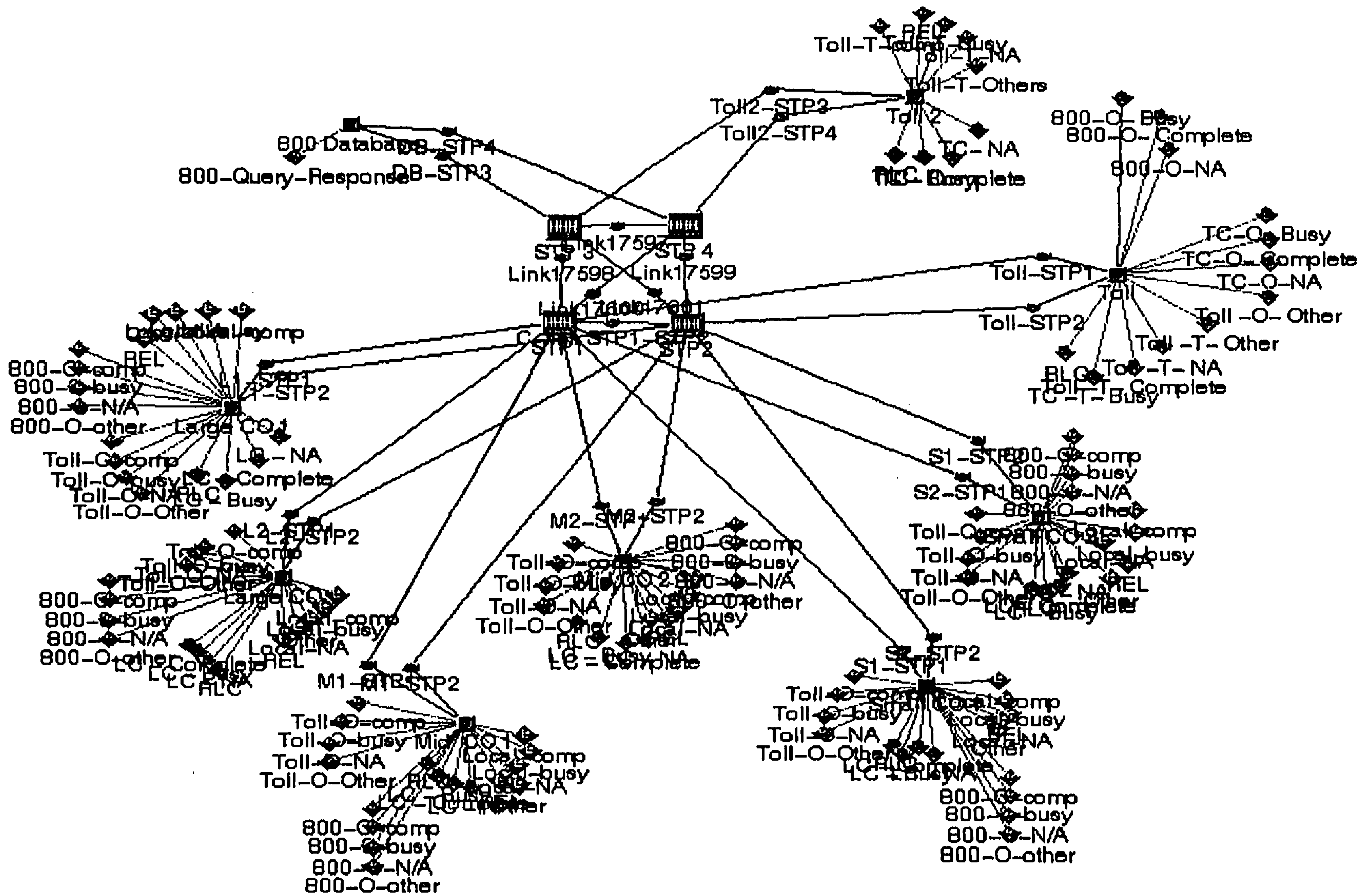


	Link Utilization			Transmission Delay (ms)		
	Normal	E02-STP1 Cut	E01-STP2 & E02-STP1	Normal	E02-STP1 Cut	E01-STP2 & E02-STP1
C-link STP1->STP2	0%	0%	18.63%	0	0	1.86
Clink STP1<-STP2	0%	0%	18.06%	0	0	1.803
E01-STP1->CO	9.11%	0%	18.06%	1.818	0	1.803
E01-STP1<-CO	9.20%	0%	18.63%	1.836	0	1.86
E01-STP2->CO	8.92%	18.01%	0%	1.781	2.326	0
E01-STP2<-CO	9.46%	18.67%	0%	1.888	2.405	0
E02-STP1->Other Cos	9.20%	0%	0%	1.836	0	0
E02-STP1<-Other Cos	9.11%	0%	0%	1.818	0	0
E02-STP2->Other Cos	9.46%	18.67%	18.63	1.888	2.405	1.86
E02-STP2<-Other Cos	8.92%	18.01%	18.06	1.781	2.326	1.803

Toll calls and 800 calls (TCAP)



Model for Toll and 800 Calls

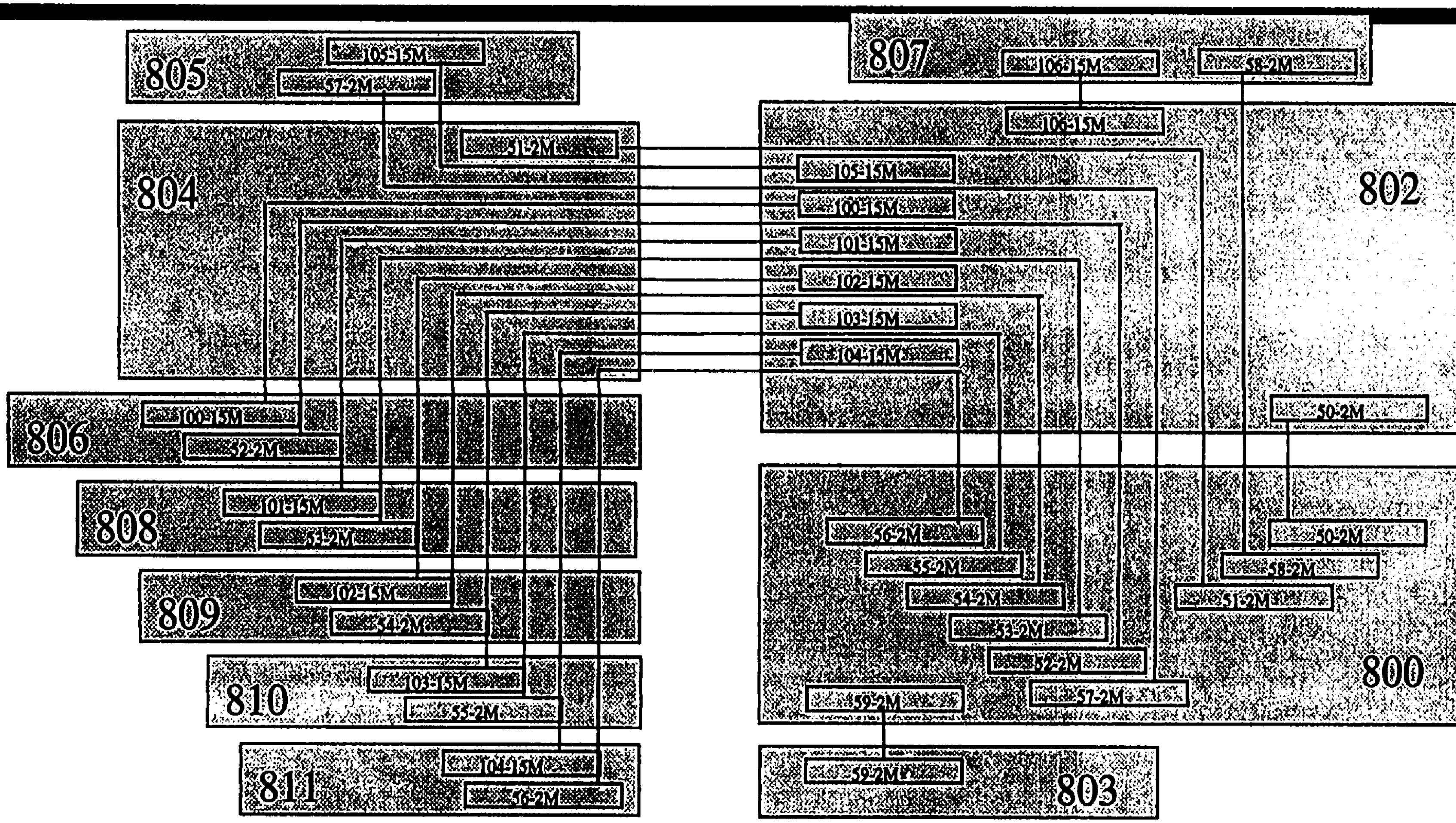


ATM Network:
Study of the link utilization

ATM Network

- **What is ATM?**
- **Study the efficiency of the MTS ATM Network**
- **Base of the physical network structure to study the behaviour of the network with PVC and SVC connections.**

MTS ATM Network Topology



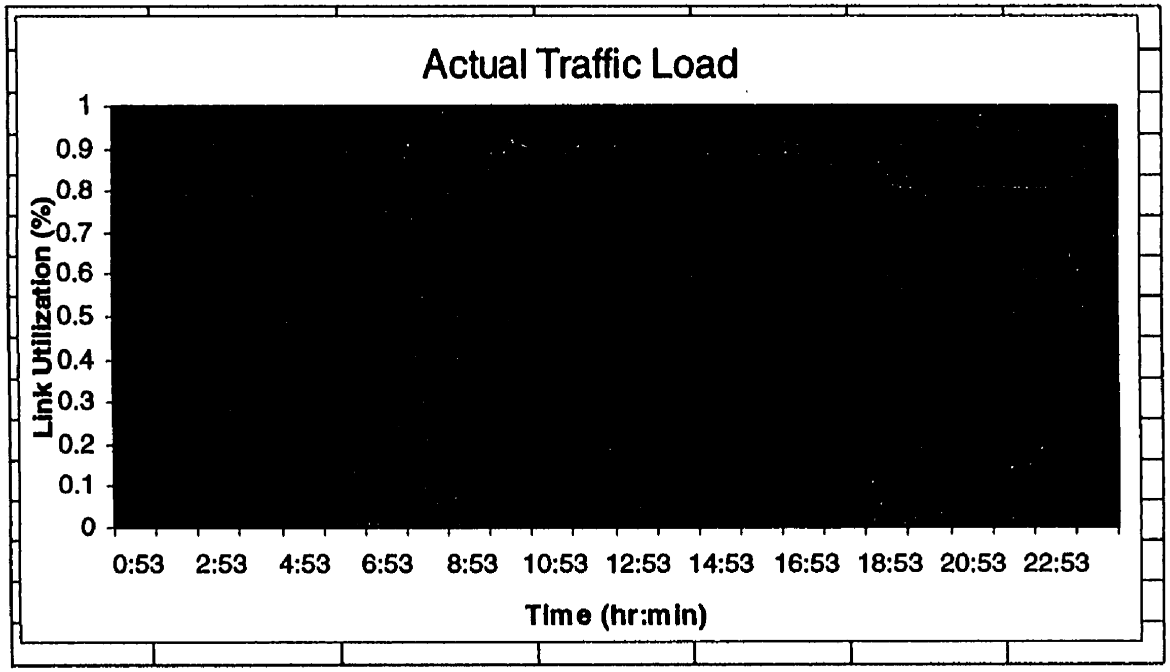
Service	800	802	803	804	805	806	807	808	809	810	811
Ethernet	✔					✔	✔				
WAN		✔									
ATM		✔			✔	✔	✔	✔		✔	✔
T3		✔		✔		✔	✔	✔	✔	✔	✔

Actual Traffic Load on the MTS ATM Network

Allocated

Link	BW (Mbps)	Link	BW (Mbps)
811⇔810	17	805⇔804	17
810⇔809	34	804⇔802	102
809⇔808	51	807⇔802	17
808⇔806	68	800⇔802	20
806⇔804	85	803⇔800	2

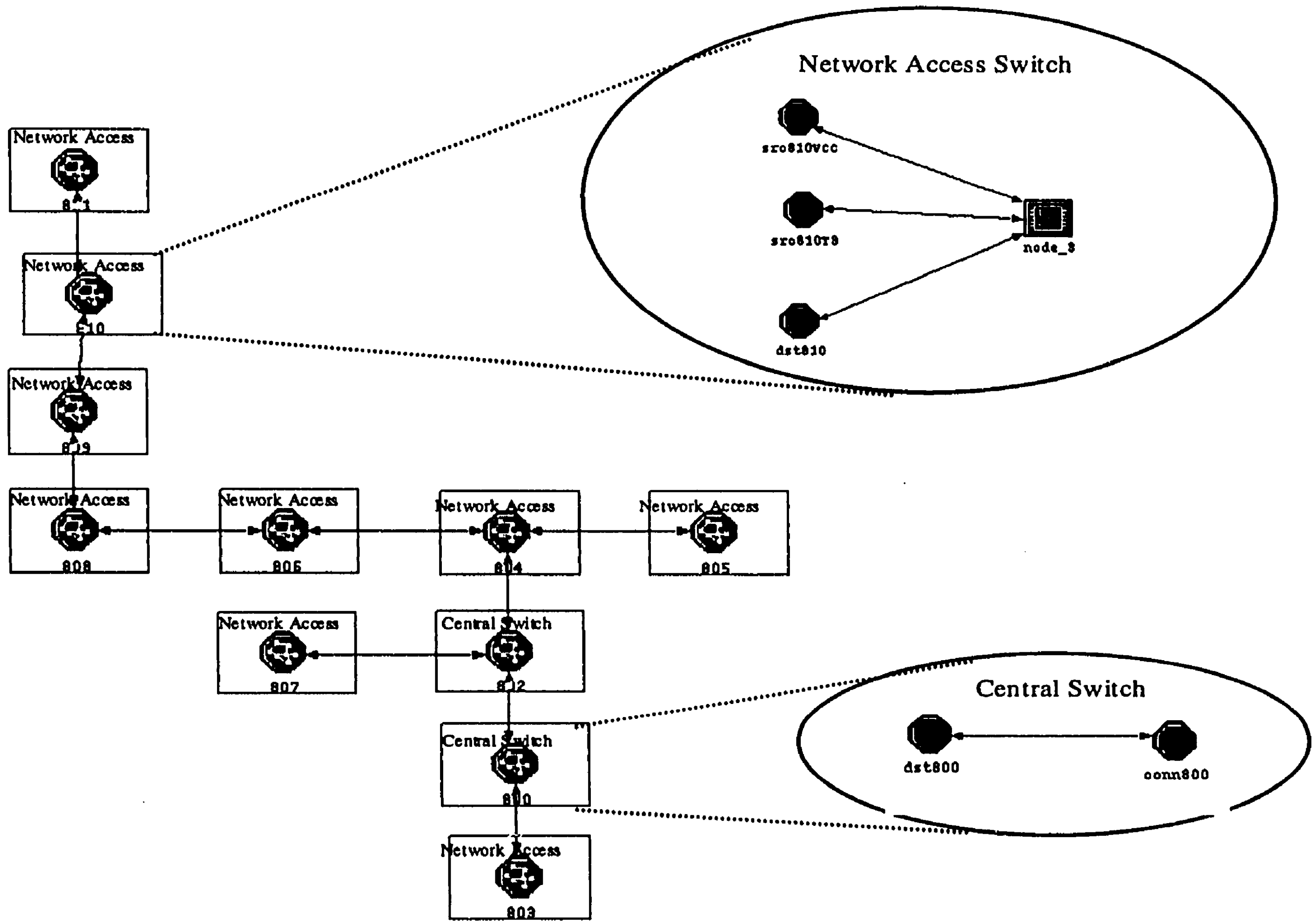
Measured



PVC Simulation Model Assumptions

- OC-3 speed
- Single link connections
- Switch 800 and 802 are central switches and also the destination switch
- Each switch provides a VCC and a T3 services with Class D traffic
- Traffic generation is deterministic
- Size of the packet is fixed.

PVC Network Layout



Network Parameters

- CBR like traffic
  - Deterministic interarrival time
  - For validation and simulates the worst case scenario
- PDU size = 1000 bits ~ 3 ATM cells using AAL5
- Simulation run = 100 seconds long
- Interarrival time = {0.01, 0.02, 0.05, 0.1, 0.2, 0.5} sec
- traffic generation is on/off with P=0.5
- 2 source nodes per switch

PVC Network Simulation Results

Link	Number of Source Node	Estimated Link Utilization (bps)	Recorded Link Utilization (bps)	Percentage Error (%)
811⇔810	2	2544	3600	41.5
810⇔809	4	5088	6001	17.9
809⇔808	6	7632	8483	11.2
808⇔806	8	10176	10735	5.5
806⇔804	10	12720	12953	1.8
805⇔804	2	2544	3723	46.3
804⇔802	14	17808	17569	1.3
807⇔802	2	2544	2807	10.3
800⇔802	18	22896	19825	13.4
803⇔800	2	2544	1187	53.3

Interarrival time = 0.5 sec

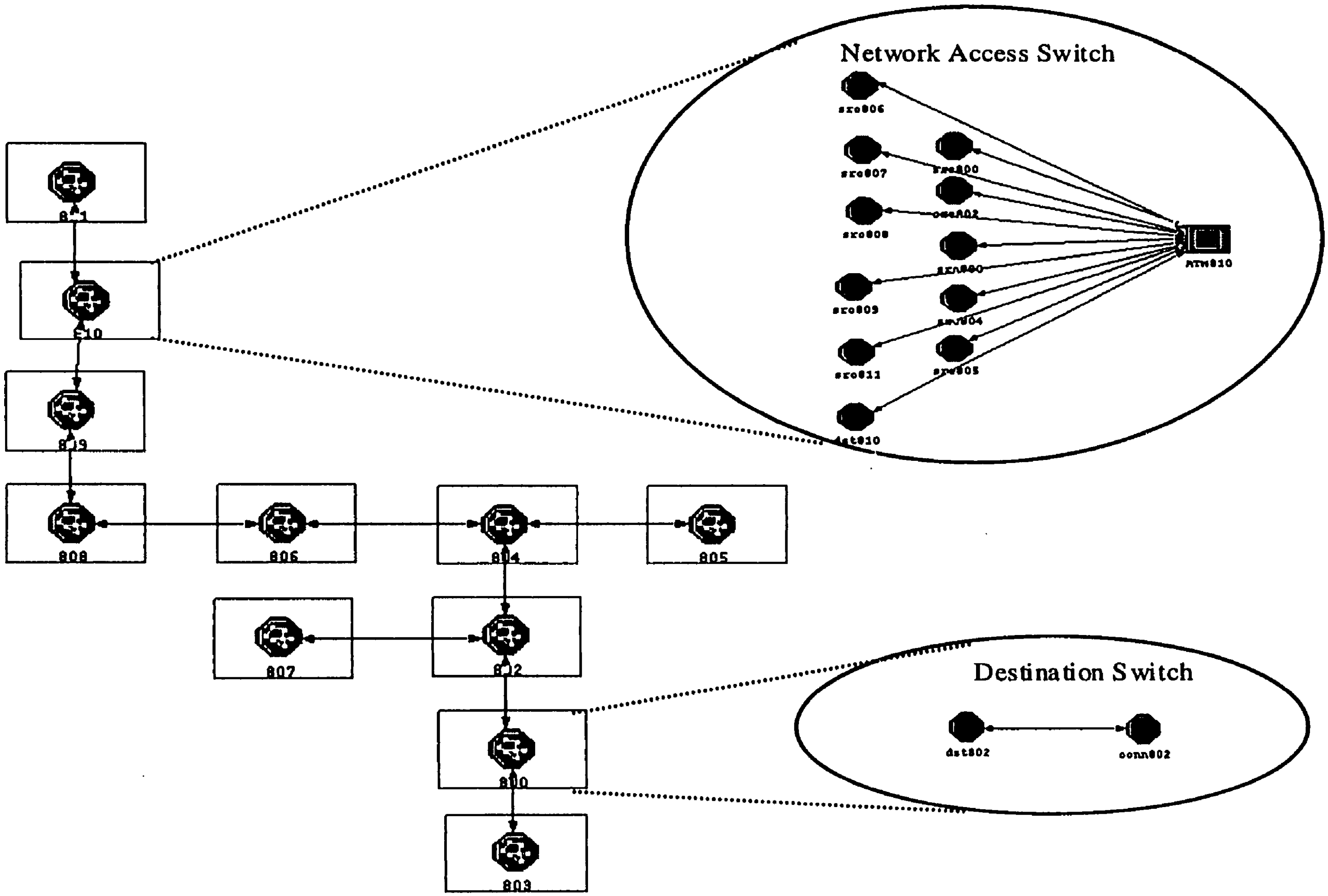
Different simulation run at the bottle neck, 800<=>802

Interarrival Time (sec)	Estimated Link Utilization (bps)	Simulated Link Utilization (bps)	Percentage difference (%)
0.01	1144800	889439	22.31
0.02	572400	464302	18.88
0.05	228960	187450	18.13
0.1	114480	92667	19.05
0.2	57240	46024	19.60
0.5	22896	19825	13.41

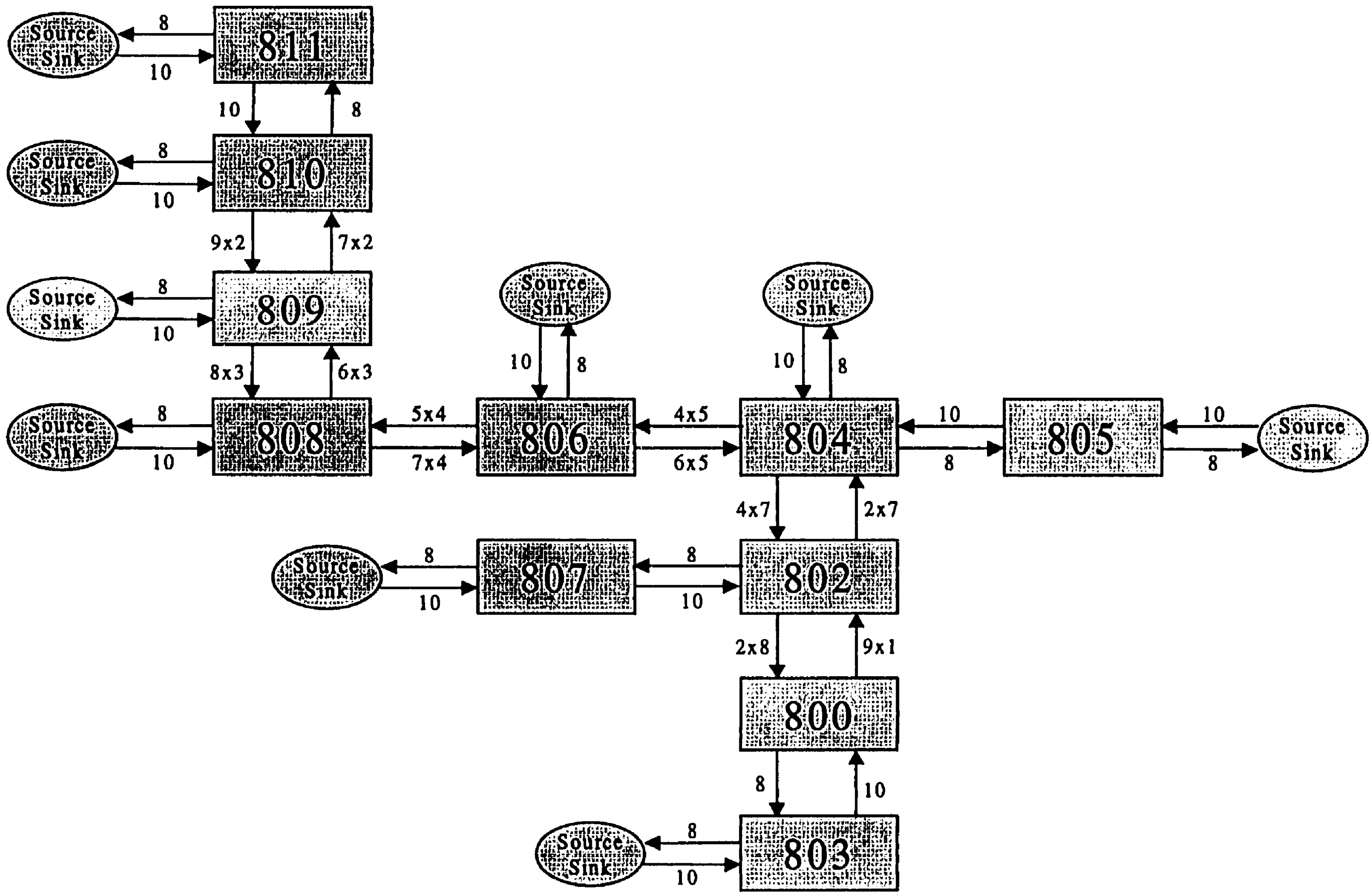
SVC Simulation Model Assumptions

- The bottle neck is at 804<=>806
- OC-3 speed with single link connection
- All switches are source and destination switch, except 800 and 802 are destination only
- Each source switch has 10 source nodes and one destination node
- Traffic generation is deterministic
- Data packet size is fixed
- Same parameters as the ones used in the PVC model

SVC Network Layout



SVC Load



811
810
809
808
806
804
805
807
802
800
803
Source Sink
8
10
9x2
7x2
8x3
6x3
5x4
7x4
4x5
6x5
4x7
2x7
2x8
9x1

SVC Network Simulation Results

Link	Up/ Right traffic units	Down/ Left traffic units	Total traffic units	Estimated link utilization	Recorded link utilization	Percentage difference (%)
811⇔810	10	8	18	228960	196524	14.17
810⇔809	18	14	32	407040	356949	12.31
809⇔808	24	18	42	534240	486108	9.01
808⇔806	28	20	48	610560	545022	10.73
806⇔804	30	20	50	636000	563361	11.42
805⇔804	8	10	18	228960	195371	14.67
804⇔802	28	14	42	534240	441688	17.32
807⇔802	10	8	18	228960	230026	0.47
800⇔802	16	9	25	318000	307341	3.35
803⇔800	8	10	18	228960	227557	0.61

Interarrival time = 0.05 sec

Different simulation run at the bottle neck, 804<=>806

Interarrival Time (sec)	Estimated Link Utilization (bps)	Final Link Utilization (bps)	Percentage difference (%)
0.02	1590000	1369457	13.87
0.03	1060000	792199	25.264
0.04	795000	748727	5.82
0.05	636000	563361	11.42
0.1	318000	272565	14.28
0.2	159000	151080	4.98
0.5	63600	69091	8.63
1	31800	39450	24.06
2	15900	26327	65.58
5	6360	18438	189.9
10	3180	16264	411.44
50	636	13609	2039.78

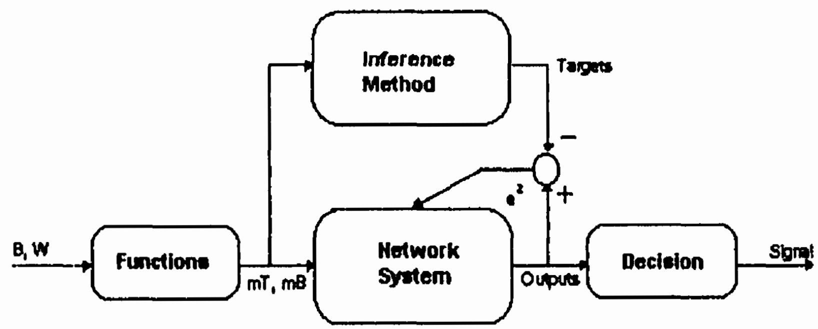
Comparing PVC vs. SVC Performance

- **Limitation: link capacity**
- **SVC network link capacity = 2.77 PVC network**
  - Even though the traffic is deterministic, but the network resources are not dedicated for idle connections.
- **Trade-offs:**
  - SVC require more computational power
  - SVC need a better software for accounting
  - PVC simple but needs a bigger pipe

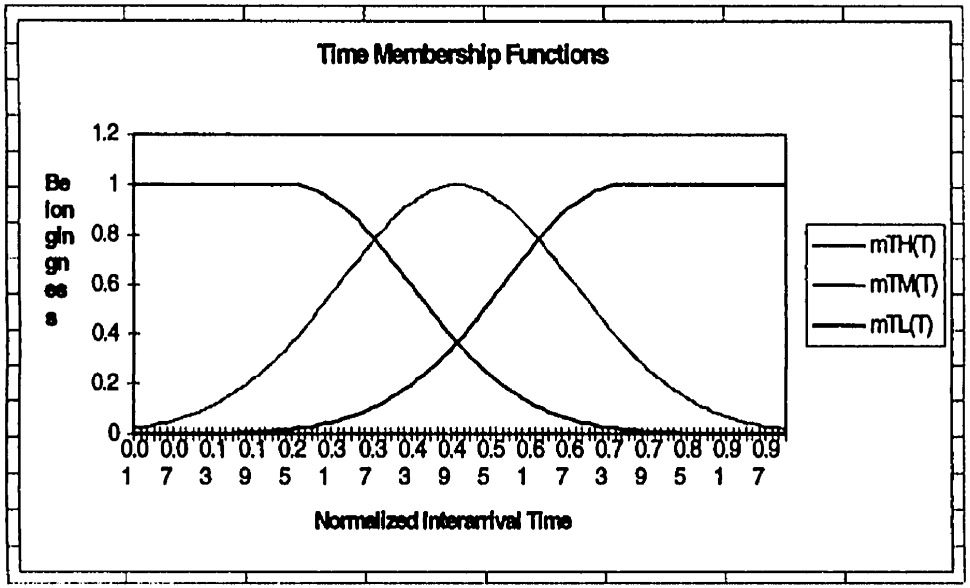
Congestion Control Mechanism

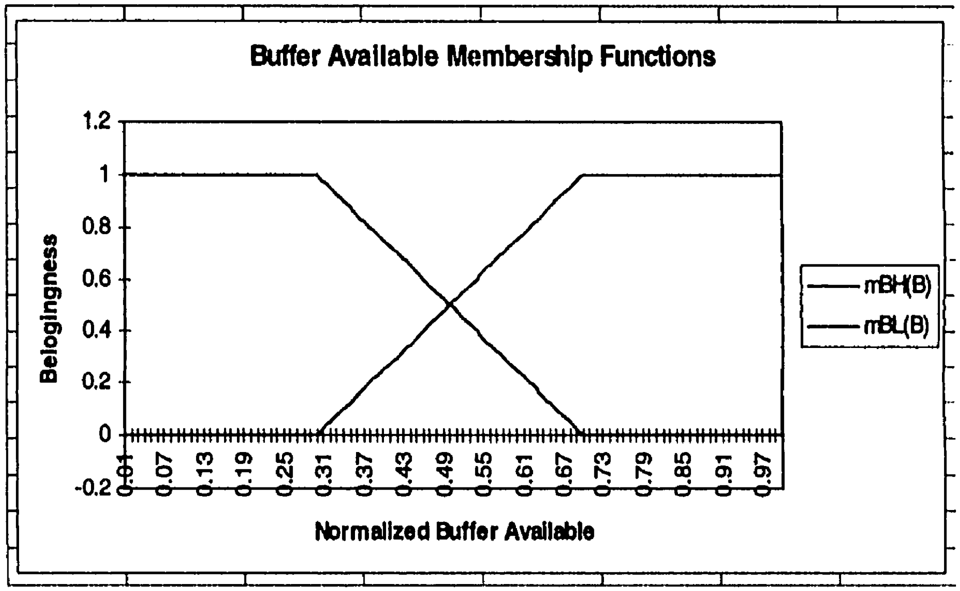
Congestion Control Mechanism for ABR Traffic

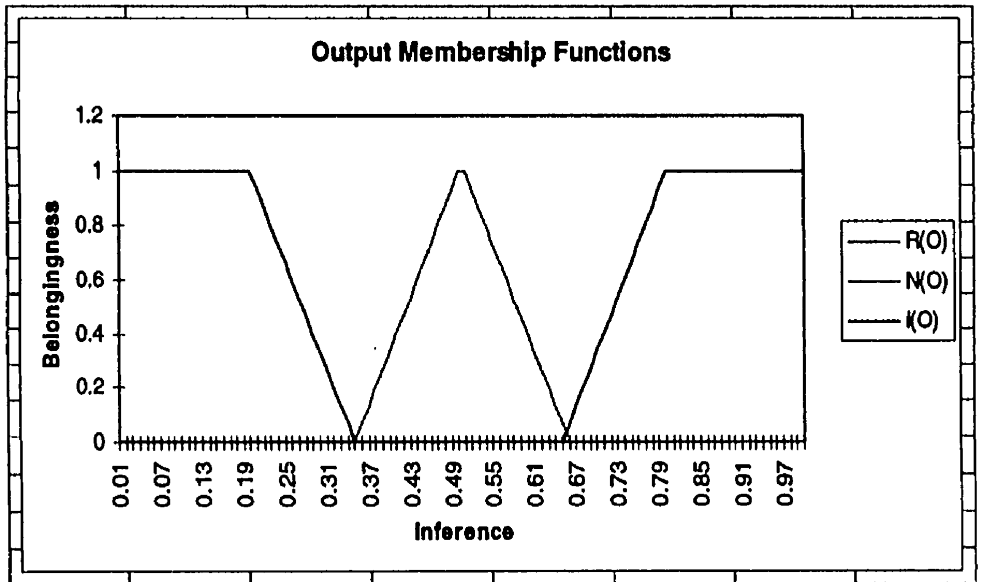
- ABR uses feedback mechanism for flow control
- New mechanism based on fuzzy set theory
  - inference method
  - based on the interarrival time and current buffer size to determine the belongingness of the condition
- The approximation methods:
  - Analytical method
  - Fuzzy Neural Network (FNN)
  - Artificial Neural Network (AN N)
- FNN and ANN
  - with supervised learning
  - with gradient-descent feedback



Membership Functions

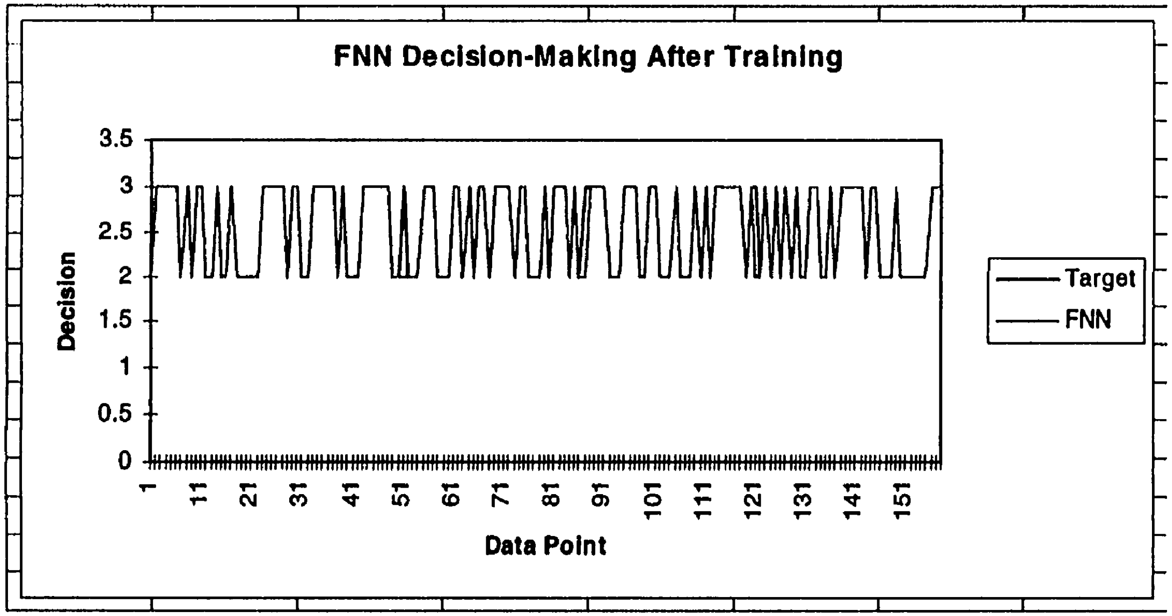






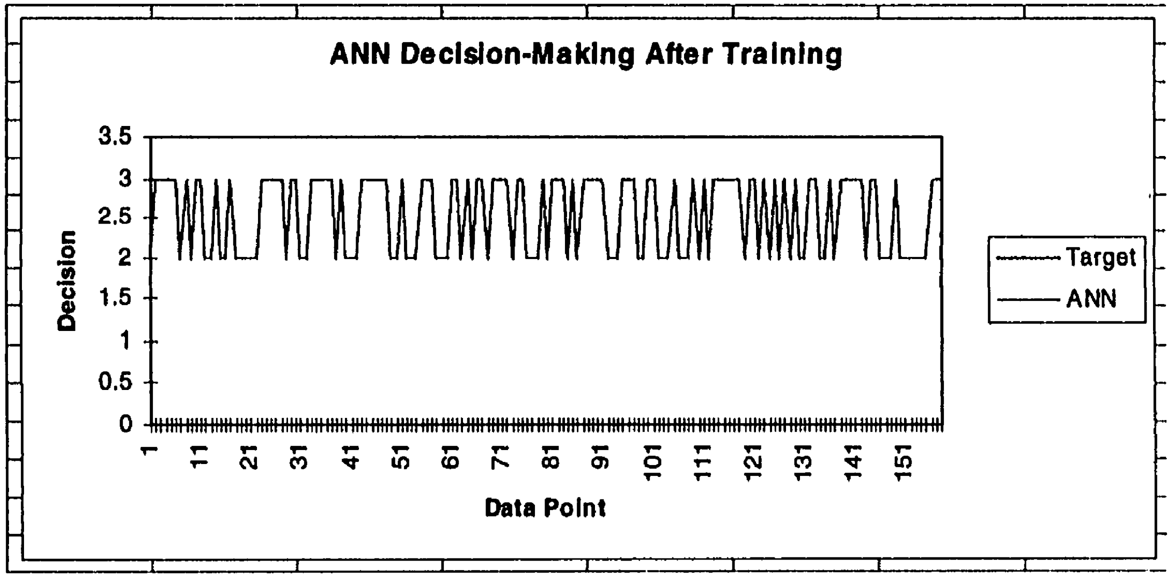
	mTH	*mTM*	*mTL*
mBH	*R*	*R*	*N*
mBL	*R*	*N*	*I*

Decision-Making after the Learning



1 = reduce
2 = no change
3 = increase

- ANN requires less time to compute
- ANN has better overall performance



Summary

- **Introduction to network modeling**
  - modeling techniques
  - commercial simulation tools
- **A few models were presents to study the adequacy of the tools for network modeling**
  - potential problems associated with the traffic generators
  - component libraries
- **CCS7 network simulation**
- **ATM PVC network vs. SVC network simulation**
- **A new ABR congestion control mechanism**

Conclusions and Recommendations

- **Conclusions:**
  - Commercial simulation tools can be used for network modeling, but may not fit all cases
  - Users may need to validate the libraries before using them
  - There are pros and cons in both tools
- **Recommendations**
  - caution would be recommended when using commercial tools
  - perform an analytical study if possible (tools do not necessarily provide an easier solution)
  - it is better to study complex models with the help of tool
  - not recommended to use the same tool for all kinds of studies
  - perform domain analysis first, before modeling the network in detail