# Chapter 1

# Introduction

# Chapter 1: Introduction

This chapter explains briefly ATM technology. QoS in ATM and the different categories of ATM services are explained too. Basic concepts of dynamic ATM simulation is introduced in 1.3 followed by some research on various popular network simulation tools.

## 1.1 Introduction to Asynchronous Transfer Mode (ATM)

Asynchronous Transfer Mode (ATM) is a switching, multiplexing technology over a B-ISDN network. ATM network is a high-speed network that transports small 53-bytes fixed length cells for communication. Each 53 bytes cell contains 5 bytes header and 48 bytes payload. The use of small cells may reduce queuing delay and therefore increase the throughput. The fixed length cells can be switched more efficiently, which is an important switching mechanism in computer hardware.

The network service provided is connection-oriented by establishing a virtual connection (VC) between a source and destination. The VC is dedicated to the pair of source and destination, this speed-up the data transfer rate. ATM is a network that can achieve gigabit-per-second data transfer rate. More than one VC can be established through a physical link. The VCs are referred as virtual channel connections (VCCs). VCCs that have same end points are grouped together to become a single virtual path connection (VPC). Each VCC has virtual channel identifier (VCI) and pair with a virtual path identifier (VPI), the VCC will have a unique connection identifier.

Switches are used to connect ATM host and networks. Each switch contains a routing table that consists of in/out pair of switch port and connection identifiers. Switching happens in the switches where incoming channels will be mapped to appropriate outgoing channels.

An attractive feature of ATM network is that it can support high bandwidth demand real-time application and multimedia transportation. Through Quality of Service (per-connection basis), which negotiated during connection set-up, ATM will reserve resource to guarantee the traffic of application during the connection time. Small fixed length cells feature enables the integration and transport of data, voice, video and other multimedia data over the same network backbone.

## 1.2 Quality of Service (QoS) in ATM

Quality of Service (QoS) can be defined as a basic set of requirements that the network must meet in order to deliver application traffic in a usable form. ATM supports various QoS per-connection basis. An end-system can request a certain QoS from the network when requesting a connection to be set up. After the QoS be granted, the QoS will be guaranteed throughout the connection duration.

The following QoS parameters are related to the source traffic characteristics:

- Peak Cell Rate (PCR): The maximum cells that can be transmitted at one time by a source on an ATM connection.
- Sustainable Cell Rate (SCR): The maximum average rate that can be reached on an ATM connection over a long interval. It is useful only if SCR < PCR.
- Cell Loss Ratio (CLR): The percentage of cells that are lost in the network due to error and congestion and are not delivered to the destination successfully. Each cell has a "Cell Loss Priority (CLP)" bit in the header. CLP=1 cell will be dropped prior to CLP=0 cell during network congestion.
- Cell Transfer Delay (CTD): This is the delay experienced by a cell between the point it enters the network and exits the other end of network. It includes propagation delays, queueing delays at various intermediate switches, and service time at queueing points.
- Cell Delay Variation (CDV): This is a measure of variance of CTD. High variation implies larger buffering for delay sensitive traffic such as voice and video.
- Cell Delay Variation Tolerance (CDVT) and Burst Tolerance (BT): A small range of delay variation allowed in the inter-cell time. CDVT enforce PCR and BT enforce SCR.

- Maximum Burst Size (MBS): The maximum number of cells that can be transmitted continuously at the PCR without violating the SCR.
- Minimum Cell Rate (MCR): The minimum cell transmission rate desired by a source, and thus can be zero in value.

There are six categories of ATM services:

- Constant Bit Rate (CBR)

  The CBR is used by connection that requires a static bandwidth amount to be allocated throughout connection lifetime. The connection is characterized by Peak Cell Rate (PCR), which means the maximum cells to be transmitted at one time. The source may transmit cell at, or below the PCR. This service category is intended to support real-time application such as voice, video, and circuit emulation. All this applications require constant bit rate traffic to assure the minimum delay during transmission over network.

- Real-Time Variable Bit Rate (rt-VBR)

  Connection that uses rt-VBR will have various transmission rates during its lifetime. rt-VBR connections are characterized in terms of a Peak Cell Rate (PCR), Sustainable Cell Rate (SCR), and Maximum Burst Size (MBS). If delay of a cell beyond the value specified by Maximum Cell Transfer Delay (maxCTD), then the performance of application can be affected. rt-VBR may support applications such as variable bit rate video compression.

- Non-Real-Time Variable Bit Rate (nrt-VBR)

  nrt-VBR is almost similar to rt-VBR except that no delay bounds area is associated with this service category. Instead, the application expects a low cell ratio. The nrt-VBR is intended for non-real-time applications such as statistical multiplexing of connection.

- Available Bit Rate (ABR)

  In ABR service category, bit transfer rate of connection depends on the traffic condition of the transmission network. A flow control mechanism is specified to support several types of feedback to control source rate. The feedback is conveyed to the source through Resource Management Cells (RM-cells). With the feedback, an end-system is expected to adapt its traffic accordingly to experience a low cell loss ratio and obtain a fair share of available bandwidth based on a network specific allocation policy. When establishing an ABR connection, the peak cell rate (PCR) and minimum cell rate (MCR) should be specified by the end-system to the network. The ABR service is not intended to support real-time application, as it does not require bounding the delay.

- Unspecified Bit Rate (UBR)

  The UBR does not specify any traffic related service guarantees. The source is free to send data up to a specified maximum amount. The network will not guarantee on the cell loss rate, and delay may happen. It is intended for non-real-time applications. Examples of such applications are traditional computer communications application such as file transfer and email.

- Guaranteed Frame Rate (GFR)

  The service guarantee of GFR is based on AAL-5 PDUs (frames). Under the congestion conditions, the network attempts to discard complete PDUs instead of discarding cells without reference to frame boundaries. On establishment of GFR connection, the end-system specifies a PCR, and a MCR that is defined along with a Maximum Burst Size (MBS) and a Maximum Frame Size (MFS). It is intended to support non-real-time applications that may require a minimum rate guarantee and can benefit from accessing additional bandwidth dynamically available in the network.

Connections are categorized into these service categories which, relate the QoS requirements to network behavior. The ATM service categories uses the following QoS parameters:

- Peak-to-peak Cell Delay Variation (Peak-to-peak CDV)
- Maximum Cell Transfer Delay (maxCTD)
- Cell Loss Ratio (CLR)

CTD is defined as the elapsed time between a cell exit event at source and the entry event at the destination of the corresponding cell for a particular connection. Fixed delay is the minimum delay that caused by propagation through the physical media, delay caused by transmission network, and fixed components of switch processing delay. On the other hand, CDV is the variable portion of delay. Factors that influence CDV includes buffering, cell scheduling, and the variable components of switch processing delay. CLR is the ratio of lost cells to total transmitted cells on a connection. The CLR is a function of a physical link's error rate and the congestion management algorithms used in an ATM switch.

Peak-to-peak refers to the difference between the best and worst case of CTD. The best is where CTD is equal to fixed delay (minimum delay). Whereas, the worst case is equal to value likely to be exceeded with probability no greater than $\alpha$. The peak-to-peak CDV is the difference between the $(1-\alpha)$ quantile of the CTD and the fixed CTD that a delivered cell may experience during connection lifetime. The maxCTD defined the $(1-\alpha)$ quantile of CTD for a connection. The CLR at connection request time is used to place an upper bound on $\alpha$.

## 1.3 ATM Network Simulation

Dynamic simulation of ATM network can provide an environment for researchers to study and evaluate ATM network performance with minimum cost. It is also a tool for ATM network planning. By using ATM simulator, network planner can design the network infrastructure with different network configuration and traffic load to make performance analysis and measurement on the simulated network. With the information obtained, the network can be geared for optimal performance.

The ATM network simulation requirements can be categorized into the following aspects:

- ATM Traffic Modeling

  There are three basic traffic types need to be modeled: video traffic, Internet traffic of WWW, and aggregate Ethernet LAN traffic. All the traffic models should capture different communication service behaviors demanded by the users. The traffic model will generate ATM cells as input for network.

- ATM Switch and Network Modeling

  Network modeling involves protocol and link modeling. Protocols need to be modeled includes ATM switch protocols, call setup and release protocols, and policing and admission control protocols. Links have to be modeled implicitly based on the flow of cells in the network instead of explicitly as a separate component. Switch models should be able to reflect the characteristics of cell flow, which is determined by the architecture of switch type.

- Simulator Performance

  The simulation performance will be judged base on the cell characterization and simulation run time. Normally, an application will be required simulations of between $10^9$ and $10^{12}$ cells. The desirable simulation run times are less than two hours with a maximum of twelve hours for overnight execution.

- Production versus Research Issues

  Since the simulator will serve research purposes, the production of the simulator are to be related to the research issues. Some issues of interest in ATM network include: workload, traffic control, congestion control, and transmission rate.

## 1.4 Network Simulation Tools

### 1.4.1 NIST ATM/HFC Network Simulator

The ATM/HFC was developed at the National Institute of Standards and Technology (NIST) to provide a flexible testbed for studying and evaluating the performance of ATM and HFC networks. It provides a graphical user interface that gives user an interactive modeling environment. User may create different network topologies, control component parameters, measure network activity, and log data as input. Various performance measures can be displayed in graphic/text form on the screen or saved to files.

The simulator is event driven. Components send each other events in order to communicate and send cells through the network. All the events will be maintained in an event queue sorted by time. To trigger an event, the global will be set to the time of the first event and any action scheduled to take place is undertaken. The time of event can be set at current time or any time in the future.

The ATM network components that will take place in simulator includes:

- ATM Switch

  This component is responsible to switch or route the cells from source to destination over several virtual channel links. After receive an incoming cell, the switch will determine its outgoing link by looking in the routing table. If the outgoing link is busy, the cell will be queued, otherwise cell will be sent through the particular link.

- Broadband Terminal Equipment (B-TE)

  A B-TE is a broadband ISDN node such as host computer, workstation, etc. It has one or more ATM Application on one side and a physical link on the other side. A cell sent by the Application side is forwarded to the physical link. If the link is busy, the cell will go into a queue.

- ATM Application

  The ATM Application can be considered as traffic generator for the network. The applications include UBR application, CBR application, VBR MPEG, and TCP/IP. For all application types, the start time and number of megabytes to be sent should be specified.

- Physical Link

This component simulates the physical media (fiber optic or copper wire) on which cells will be transmitted. The link speed and link length will be specified and link utilization will be produced in terms of bit rate (Mbits/s).

### 1.4.2 Netsim

Netsim is a simulation program for the study of Ethernet Medium Access Control layer protocol performance for Local Area Network (LAN). Netsim allows the user to investigate the effects of the packet arrival process, the size of packets transmitted, the speed of the network, and the layout of the network on performance. Users are allowed to configure the number of hosts, their positions on the network, and the overall length of the network. The simulator provides results on different performance metrics, such as overall throughput, average packet delay, variance of packet delay, and number and distribution of collisions.

The Netsim simulator is a single process discrete event simulator for packet-switched network. The same format of output will be produced and written to provide as identical a simulated operating environment for the protocol as possible. Individual stations does not model packet buffering, it is assumed to generate a new arrival only after their previous packet has been transmitted successfully. A single event queue is used for the simulation. At any given time, the event queue contains one entry for each station on the network. The entry contains the simulation time of the next transmission attempted by the station and the simulation time at which the packet was originally submitted for transmission and other protocol specific information as well.

Main features of Netsim are as followings:

- Finite population simulation

  Netsim simulates a user specifiable finite population of stations on a network. This gives a more accurate picture of network behavior than simulation that rely on an infinite population model. Infinite population model makes analytic models more mathematically tractable and simulation model simpler at the expense of accuracy.

- Flexible output process specification

  Each station produces output process or sequence of interpacket time and packet lengths that can be specified in several ways. For example, they can be generated randomly from traditional probability distribution such as the exponential and uniform distribution, or can be deterministic that is always be the same value. This flexibility allows a wide variety of network traffic loads to be recreated.

- User oriented delay statistics

  The simulator keeps track of the queueing delay experienced by transmitted packets. Queueing delay is defined as the difference between the time the packet was originally submitted for transmission and the time at which it completes successful transmission. This is more useful in determining the network response time from user point of view than the network-oriented metric of overall inter-arrival time.

- Automatic offered load calculation

  The offered load can be automatically calculated from the description of individual station to be simulated.

- Automatic experiment repetition

  A simulation experiment can be repeated several times starting with different random number generator. The average of the collected statistics is then used for data analysis to produce more reliable results.

- Varying parameter

  The varying parameter mechanism enables a sequence of simulation experiments to be run by varying a single parameter without human intervention.

### 1.4.3 REAL

REAL is a simulator for studying the dynamic behavior of flow and congestion control schemes in packet switch data network. By taking the scenario, which is a description of network topology, protocols, workload, and control parameters as input. It can produce statistic such as the number of packets sent by each source of data, the queuing delay at each queuing point, the number of dropped and retransmitted packets sent by each and other similar information as output. Scenarios are designed using NetLanguage, a simple ASCII representation of the network.

The network is modeled as a graph, where nodes (vertices) represent either sources or sinks of data, or gateways. The interconnection between the nodes is the topology. At each source, network wide parameter, the transport protocol (in particular, the flow control), and the workload at each source need to be specified. Control parameters such as the latency, the size of trunk board buffers, packet sizes and etc must be specified also. Each user is modeled as a source of data regulated by a flow control protocol. The combination of a workload and flow control protocol is implemented by a single function, which is executed in parallel by the underlying thread-based simulation package such as NEST (NEtwork Simulation Testbed), can be thought of as being an independently scheduled and pre-emptable entity.

The sources can be categorized into two types: flow controlled and non-flow controlled data sources. Flow controlled sources use acknowledgement and timeout to implement reliable transport layer functionality over a lossy network. Non-flow controlled sources generate data from a known distribution and do not provide a reliable transport functionality. Flow controlled sources assume that the user have a finite number of data packet ready to transfer, and that this data is available for transmission without any delay. So, whenever the flow control protocol allows data transmission, a packet to transmit would always be available.

Gateway is the router, which relay the data packet on the network during transmission. They are categorized by the scheduling discipline such as first come first served, fair queueing, etc. Sinks receive data packet from any source. If the source is a guarantted service host, the sink only collects statistics, and does not send an acknowledgement. Otherwise, it keeps track of the highest in-sequence packet received thus far, and sends an acknowledgement with that sequence number.

### 1.4.4 OPNET

OPNET is the first commercial network simulator, which was originally developed at MIT. It allows the definition of a network topology, the nodes, and the links that go towards making up a network. The processes that may happen in a particular node can be user defined, as can the properties of the transmission links. A simulation can then be executed, and the results analysed for any network element in the simulated network. OPNET consists of three main tool categories: Model Development, Simulation Execution, and Result Analysis. There are tools under each main tool category.

Model Development includes:

- Network Editor – used to design the network models with different nodes connected.
- Node Editor – used to place the models of the nodes that consist of modules used in the network.
- Process Editor – used to define the processes that run inside the node modules.
- Parameter Editor – allows the definition of parameters used in the node modules and process model.

Simulation Execution Tools includes:

- Probe Editor – used to place probes at various points of interest in the network model. These probes can be used to monitor any of the statistics computed during simulation.

- Simulation Tool – allows the user to specify a sequence of simulations, along with any input and output options, and any different runtime options.

Result Analysis Tools includes:

- Analysis Tool – will display the results from a simulation or series of simulations as graphs.
- Filter Editor – used to define filters to mathematically process, reduce, or combine statistical data.

Main features of OPNET are as followings:

- Hierarchical network models – Manage complex network topologies with unlimited sub-network nesting.
- Object-oriented modeling – Nodes and protocols are modeled as classes with inheritance and specialization.
- Clear and simple modeling paradigm – Model the behavior of individual objects at the process level and interconnect them to form devices at the "Node Level"; interconnect devices using links to form network at the "Network Level".
- Finite state machine modeling of protocols and other processes.
- Comprehensive support for protocol programming – 400 library functions support and simplify writing protocol models.

### 1.4.5 OMNET++

OMNeT++ is another commercial network simulator. OMNeT++ was written with a large OPNET experience in mind. The intent was to create a tool that offers comparable features yet supports advanced concepts like object-oriented programming, interactive (GUI) execution environment, and provides more open interfaces. It is primarily designed to simulate computer networks, multi-processors and other distributed systems, but it may be useful for modeling other systems too. OMNeT++ implements a system consists of a number of entities (we call them *modules*) communicating by exchanging messages; modules can be nested, that is, several modules can be grouped together to form a bigger unit (a *compound module*).

OMNeT++ is an object-oriented modular discrete event simulator. It can be used for modeling communication protocols, computer networks and traffic modeling, multiprocessor and distributed systems, and any other system where the discrete event approach is suitable.

An OMNeT++ model consists of hierarchically nested modules. The depth of module nesting is not limited, which allows the user to reflect the logical structure of the actual system in the model structure. Modules communicate with message passing. Messages can contain arbitrary complex data structures. Modules can send messages either directly to their destination or along a predefined path, through gates and connections. Modules can have parameters that are used for three main purposes: to customize module behavior; to create flexible model topologies; and for module communication, as shared variables.

Modules at the lowest level of the module hierarchy are to be provided by the user, and they contain the algorithms in the model. During simulation execution, simple modules appear to run in parallel, since they are implemented as co-routines. To write simple modules, users do not need to learn a new programming language, but they are assumed to have some knowledge of C++ programming. OMNeT++ simulations can feature different user interfaces for different purposes: debugging, demonstration, and batch execution. User interfaces also facilitate demonstration of how a model works.

### 1.4.6 INSANE

INSANE is a network simulator designed to test various IP-over-ATM algorithms. Internet protocols supported include large subsets of IP, TCP, and UDP. In particular, the simulated TCP implementation performs connection management, slowstart, flow and congestion control, retransmission, and fast retransmit. Various application simulators mimic the behavior of standard Internet applications to provide a realistic workload such as telnet, ftp, WWW, real-time video, and real-time audio. The results of simulation are processed off-line as the simulation is expected to be large.

INSANE is an object-oriented, discrete-event simulator. The simulation core, as well as a set of primitive objects, is implemented in C++. Instances of the various primitive objects can be created and configured into complex, composite objects using the Tcl scripting language. Because Tcl provides general-purpose programming language constructs such as loops, conditions, and procedure definitions, the creation of a large simulation scenario is fairly easy. A graphical user interface facilitates easy monitoring of multiple simulation jobs running on various machines of a distributed computing cluster.

## 1.5 Introduction to Simulink

Simulink is an interactive tool for modeling, simulating, and analyzing dynamic system. It is a powerful simulation software tool that enables virtual prototypes to be built and tested. This can minimized the effort of exploring design concepts at any level of detail. Simulink provides an interactive, block-diagram environment for modeling and simulating dynamic system. It includes an extensive library of predefined blocks that can be used to build graphical models of the system using drag-and-drop operations. Supported model types include linear, non-linear, continuous time, discrete-time, multi-rate, conditionally executed, and hybrid systems. Models can be grouped into hierarchies to create a simplified view of components or subsystems. Simulink has many features that allow customization, especially with regard to incorporating existing user C code.

Simulink provides a complete set of modeling tools that can be used to develop detailed block diagram model of a system quickly. Features such as block libraries, hierarchical modeling, signal labeling, and subsystem "masking" provide a powerful set of capabilities for creating, modifying, and maintaining block diagrams. The Simulink block library consists of over one hundred blocks, providing an extensive set of standard components. Simulink provides a special type of block called S-function that allows external code (C, Fortran, or Matlab) to be incorporated into Simulink models. This feature makes Simulink an extremely flexible modeling environment. In addition to its extensive set of built-in blocks, Simulink offers features for creating customized blocks and block libraries. Besides customizing the functionality of a block, custom icons and dialog boxes for subsystems and S-function can also be created. Custom block can be saved in user-defined block library.

After studying various network simulation tools, it was decided that Simulink would be the ideal tool for dynamic modeling of cell flow for an ATM network.

## 1.6 Objective of this Project

The objective of this project is to perform dynamic modeling of cell flow for an ATM network. The cell flow is modeled for various network topologies and its performance is measured. The topologies that will be studied are:

- Single Hop topology
- Multi Hop topology
- Ring topology
- Mesh topology
- Star topology

## 1.7 Scope of this Project

- The single hop topology consists of one switch with cell flowing directly and crossed.
- The multi hop topology consists of two switches connected directly and crossed.
- Ring topology consists of three switches connected, with three types of traffic flow.

- Mesh topology consists of four switches connected, with two types of traffic flow.

## 1.8 Organization of Report

The report will be organized as followings:

**Chapter 1: Introduction**

This chapter introduces ATM network, ATM network simulation, and Simulink tool. Various existing ATM simulation tools are described. The project objective and project scope are provided.

**Chapter 2: ATM Network and Modeling**

This chapter covers the literature survey done on ATM network including ATM network topology, ATM switch, Broadband Terminal Equipment, ATM application, physical link, ATM internetworking, and traffic management and congestion control in ATM network. ATM network simulation and modeling will also be discussed.

**Chapter 3: Simulation and Simulink**

This chapter discusses the concepts of simulation and Simulink tools. Topics covered are simulation concepts, probability concepts, and queueing theory. Simulink and S-function are explained here.

**Chapter 4: Dynamic Modeling and Simulation**

The network modeling and simulation environment will be explained. Implementation issues are discussed here. The specifications of the component blocks and its algorithms are included too.

**Chapter 5: Discussions and Conclusions**

Discussions of modeling and simulation results are provided. Advantages and limitations of using Simulink in ATM simulation and area of future enhancements are highlighted here.