

Simulating ATM Technology on high-speed networks to support the training at PTIT

Phan Cong Vinh, Tran Cong Hung.
Vietnam Posts and Telecoms Institute of Technology
Tel: +84-8-8228475 Fax: +84-8-8295258
E-mail: {vinhpc,conghung}@ptithcm.edu.vn

Theme : **ISDN Technologies**

Abstract - During the research into ATM technology on high-speed networks, the lecturers and researchers into networking at Vietnam Posts and Telecoms Institute of Technology face some difficulties to be lacking in suitable simulation environments for experimenting and performance evaluating the parameters and protocol behaviors. In all the bearings if implementing on physical ATM networks is unavailable therefore the simulation tool being able to make the experimental environment is required.

In researching period we decide to choose the ATM network simulator being developed and improved at National Institute of Standards and Technology (NIST) belong to US Ministry of Industry to support the evaluation of ATM network running. It is a good visualization interface tool easy using to analyze the behaviors and status of ATM networks. It is composed of two major tools: one is engaged in designing the network topology and, another one is a protocol analyzing tool. We use that tool to design the different network architectures and analyze the anti-congestion solutions with the heterogeneous traffic sources. Particularly, we simulate the anti-congestion solution of OSU (Ohio State University) being developed by Dr. Raj

Jain and his other members with the ABR quality level traffic sources on several network topologies. The numerical and graphical results help lecturers, researchers and also senior students easy making the performance evaluations. We have applied that achievement to effectively support the research and training on networking subject at PTIT (Vietnam Posts and Telecoms Institute of Technology).

1. Introduction

The ATM/HFC Network Simulator was developed at the National Institute of Standard and Technology (NIST) to provide a flexible tested for studying and evaluating the performance of ATM and HFC networks. The ATM/HFC Network Simulator has been developed to provide a means for researchers and network planners to analyze the behavior of ATM and HFC networks without the expense of building a real network. The simulator is a tool that gives the user an interactive modeling environment with a graphical user interface.

The ATM/HFC Network Simulator allows the user to create different network topologies, set the parameters of component operation, and save/load the different simulated configurations. While the simulation is running, various

instantaneous performance measures can be displayed in graphical/text form on the screen or saved to files for subsequent analysis.

With the ATM/HFC Network Simulator, user can present some network control algorithm. In this paper, we will present our experimental results of simulating ATM technology in high-speed networks to support the training at PTIT.

The OSU scheme is also an explicit rate indication scheme simulator to the MIT scheme [1,2]. However, it does not necessarily require the switches to remember the rates of all VCs. Thus, the minimal storage requirements as well as the computational complexity becomes $O(1)$, that is, the computation or storage does not change as the number of VCs is changed. Also, it uses the exact overload as measured at the switch to determine the allowed rate.

2. Implementation & Using ATM Network Simulation Tool.

2.1. Some brief components descriptions.

Switch. This is the component used to switch or route cells over several virtual channel links.

Broadband Terminal Equipment (B-TE). This is a component to simulate a broadband ISDN node, e.g., host computer, workstation, etc.

Hybrid Fiber Coax Network (HFC). The HFC is a Cable TV (CATV) network using coax and fiber technology to provide high speed digital services to subscriber premises.

ATM application. This is a component to emulate the behavior of an ATM application at the end-point of a link.

Physical link. This component simulates the physical medium (copper wire or optical fiber) on which cells are transmitted.

2.2. The main display layout.

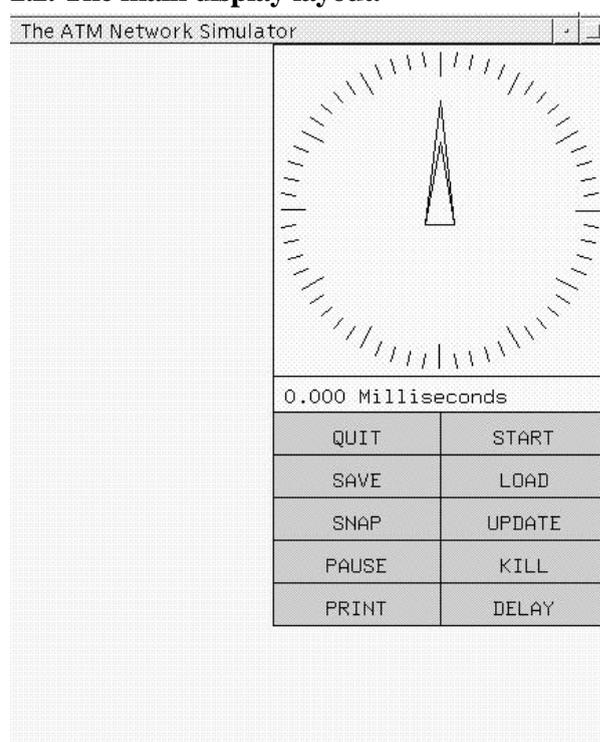


Figure 1. Main display layout

The text window

The text window appears as a bar at the bottom of the screen. The text window allows the program to present various message to user.

The Control Panel

The control panel appears on the right hand portion of the screen. It contains an analog clock, a digital clock, and an array of control buttons.

Analog clock

The analog clock indicates the passage of simulator time in a graphic style.

Digital clock

The digital clock provides a display of current simulator time accurate to the nearest 10 nanoseconds.

Control buttons

Starts: Clicking on this button will start the simulation with simulated time initialized to zero.

Pause/Run: This button toggles between two modes. When program is running the word PAUSE will be displayed. Clicking on the button will then stop all activity with all parameter and time information held in place. With the simulation stopped, the button label will change to RUN, clicking on it will cause the simulation to resume running with current settings.

Delay: This button allow the user to slow down the simulation.

Update: clicking on this button will toggle screen updating on or off.

Kill: This button may be used to stop a simulation in progress or to eliminate component.

Load: This button allows the user to load a network configuration

Save: This button allows the user to save the present configuration in a specially formatted text file which is readable by the simulator at LOAD time.

Snap: This is similar to Save, but in addition it saves the present arrangement of meter and information windows on the display.

Print: print out the network topology into PostScript file.

Quit: This is the normal exit from the simulation program.

2.3. Loading a Network Configuration.

There are three ways to specify a network configuration for the program to simulate.

- Specify the name of a *configfile* describing the network on the command line.

- Use Load command while in the simulator program.
- Create a network while in the simulator program using the tools the program provides.

2.4. Creating a Network Configuration**- Creating Components.**

Creating components (except for links) is achieved by holding the **shift** key down and clicking the **right** mouse button on the background.

- Linking Components.

ATM switches and B-TE he physical links use to connect ATM-Switches and BTE such as select the first component to be linked by clicking on it with the middle mouse button while holding down the shift key. Select the second component to be linked by the same method. At this point a line will appear and the physical link component will be created. As with other components, an information window will appear and the user will be prompted to enter a names and some input parameters.

- Creating Routes.

In the simulator, a Route is a list of adjacent components beginning and ending with ATM Applications. To create a Route, hold the shift key down and click the left mouse button on each component in the route.

2.5. Logging Data.

Data logging is a method of the recording the values of a parameter

while the simulation is running. Logging for a parameter is toggled on and off by clicking the middle mouse button on the right-hand box on the information window line for that parameter. Besides the Simulator have many functions such as: raising and lower windows, resizing windows, moving windows, modify component,...

3. OSU anti-congestion solution.

3.1. Control Cell Format.

The control cell contains the following the fields :

1. Transmitted Cell Rate (TCR).
2. The Offered Average Cell Rate (OCR) as measured at the source.
3. Rate Adjustment Factor.
4. Averaging interval.
5. The direction of feedback (backward/forward)
6. Timestamp containing the time at which the control cell was generated at the source.

The last two fields are used in the backward congestion notification option and need be present if that option is not used.

3.2. The Source Algorithm.

The source algorithm consists of three components :

1. How often to send control cells.
2. How to measure the offered average cell rate.
3. How to respond to the feedback received from the network.

These three question are answered in the next three subsections.

3.2.1. Control Cell Sending Algorithm.

The control cells are sent periodically every T interval. Although it could be done by the cell count, using interval allows the scheme to work on networks with widely varying link speeds. As shown later, the averaging interval used throughout the path should be the same. The network manager sets the averaging interval parameter for each switch. The maximum of the averaging interval along a path is returned in the control cell. This is the interval that the source uses to send the control cells. During an idle interval, no control cells are sent. If the source measures the OCR to be zero, then control cell is sent, subsequent control cells are sent, only after the rate becomes non-zero.

3.2.2. Measuring Offered Average Load.

Unlike any other scheme proposed so far, each source also measures its own load. The measurement is done over the same averaging interval that is used for sending the control cells. Notice that there are two separate parameters: transmitted cell rate and offered average cell rate. The first is the instantaneous cell rate during burst transmissions. The cells are sent equally spaced in time. The inter-cell time is computed based on the transmitted cell rate. However, the source may be idle in between the bursts and so the average cell rate is different from the transmitted cell rate. This average is called the offered average cell rate and is also included in the cell.

Normally the OCR should be less than the TCR, except when the TCR has just been reduced. In such cases, the switch will actually see a load corresponding to the previous TCR and so the feedback will correspond to the previous TCR. The OCR, in such cases is closer to

previous TCR. Putting the maximum of current TCR and OCR in the TCR field helps overcome unnecessary oscillation caused in such instances. In other words, $\text{TCR in Cell} \leftarrow \max\{\text{TCR}, \text{OCR}\}$

3.2.3. Responding to Network Feedback

The control cells returned from the network contain a “load adjustment factor” along with the TCR. The current TCR may be different from that in the cell. The source computes a new TCR by dividing the TCR in the cell by the load adjustment factor in the cell:

$\text{New TCR} \leftarrow (\text{TCR in the Cell})/(\text{Load Adjustment Factor in the Cell})$

If the load adjustment factor is more than one, the network is asking the source to decrease. If the new TCR is less than the current TCR, the source sets its TCR to the new TCR value. However, if the new TCR is more than current TCR, the source is already operating below the network’s requested rate and there is no need make nay adjustment.

Similarly, if the load adjustment factor is less than one, the network is permitting the source to increase. If the current TCR below the new TCR, the source increases its rate to the new value. However, if the current TCR is above the next TCR, the new value is ignored and no adjustment is done.

3.3. The Switch Algorithm

The switch algorithm consists of the following components:

1. How to measure the available capacity
2. How to achieve efficiency.
3. How to achieve fairness.

These issues and others arising from these are discussed next.

3.3.1. Measuring The Current Load

This consists of simply counting the number of cells received during a fixed averaging interval. The interval is set by the network manager. Based on the known capacity of the link, the switch can compute the load level and determine whether it is overloaded or underloaded. Since running a link at full load generally results in large queues, it is best to target the link utilization at close to but not quite at 100%. To achieve this the network manager selects a target utilization, say 90%. Whenever the input rate is more than 90% of the nominal capacity, the link is said to be overloaded and whenever the utilization is less than 90%, the link said to be underloaded. The link cell rate when the network is operating at the target utilization is computed:

$\text{Target Cell Rate} = (\text{Target Utilization} \times \text{Link bandwidth in Mbps})/\text{Cell size in bits}$

The current load level is then given by:

$\text{Current Load level} = (\text{Number of cells received during the averaging interval})/(\text{Target Cell Rate} \times \text{Averaging Interval})$

3.3.2. Achieving Efficiency

To achieving efficiency, all we need is to replace the load adjustment factor in each control cell by the maximum of the current load level and the load adjustment value already in the cell.

Load Adjustment Factor \leftarrow max(Load Adjustment Factor in cell, Current Load Level in this Switch)

This simple algorithm is sufficient to bring the network to efficient operation with in next round trip. However, the allocation of the available bandwidth among not be fair. To achieve fairness we need to make use of the other information in the control cells as discussed later in Section III.3.4.

3.3.3. Counting the Number of Active Sources

Like the MIT scheme, the switches in our scheme may also remember the rates declared by various source and use them in computing the fair share. However, there are two differences. First the rates declared by the source are “Offered Average Cell Rates (OCRs)”, and not the desired cell rates, which may or may not be related to the actual rates. Secondly, in the simplest version of our scheme rates of all source are not required. All we need is the number of active source, which can be counted either by counting the number of sources with non-zero OCRs or by marking a bit in the VC table whenever a cell from a VC is seen. The bits are counted at the end of each averaging interval and cleared at the beginning of each interval.

3.3.4. Achieving Fairness

In resource allocation, the top priority is to bring the network to efficient operation. Once the network is operating close to the target utilization, we need to take steps to achieve fairness. The network manager declares a target utilization band (TUB), say, $90 \pm 9\%$ or 81% to 99% . Whenever the link

utilization is in TUB, the link is said to be operating efficiently. As will be seen later, it is better to express TUB in the $U(1 \pm \Delta)$ format, where U is the target utilization level. For example, $90 \pm 9\%$ is expressed as $90(1 \pm 0,1)\%$.

Given the number of active source, the fair share is computes as follows :

Fair Share = (Target Cell Rate)/(Number of Active Sources)

To achieve fairness, we treat the underloading and overloading source differently. Underloading sources for our scheme are those sources that are using less than the fair share. While overloading source are those that using more than the fair share.

If the current load level is z , the underloading sources are treated as if the load level is $z/(1+\Delta)$ and the overloading sources are treated as if the load level is $z/(1-\Delta)$. Here Δ is the half-width of the TUB.

If the OCR in the control cell is less than the fair share, the load adjustment factor in the cell is changed as follows:

Load Adjustment Factor \leftarrow
Max(Load Adjustment Factor in the cell, $z/(1+\Delta)$)

On the other hand, if the OCR in the control cell is more than the fair share, the load adjustment factor in the cell is adjusted as follows:

Load Adjustment Factor \leftarrow
Max(Load Adjustment Factor in the cell, $z/(1-\Delta)$)

This algorithm guarantees that the system consistently moves towards more fair operation. Also, once inside the TUB, the network remains in the TUB unless the number of courses or their load pattern changes. In other words,

TUB is a “closed” operating region. These statements are true for any value of Δ less than 0.5

If Δ is small, as is usually the case, division by $1+\Delta$ is approximately equivalent to a multiplication by $1-\Delta$ and vice versa.

What Load Level Value to use?

Under highly overloaded conditions, the queues in a system may become long. The control cells may remain in the system for more than one averaging interval and the question arises as to what load level value should be used for efficiency or fairness computation. Should it be the value at the time of control cell arrival or the latest value at the time of control cell departure? The correct answer is: the value at the cell arrival time should be used. This is because the queue state at arrival more accurately reflects the effect of the TCR indicated in the control cell.

3.4. The Destination Algorithm

The destination simply returns all control cells back to the source.

3.5. Initiation Issues

When a source first starts, it may not have any idea of the averaging interval or what rate to use initially. These are two answers. First is that ATM networks are connection oriented and so the above information can be obtained during connection setup. Second, it is possible to send a control cell (with $TCR=0=OCR$) and wait for it return. When the control cell returns, it will have the information to change to the correct load level.

The initial transmission cell rate affects the network operation for only the first few (one or two) round trips. Therefore, it can be any value below (and including) the target cell rate of the link at the source. However, network managers may set any other rate to avoid startup impulses.

3.6. Unique Features of the OSU scheme

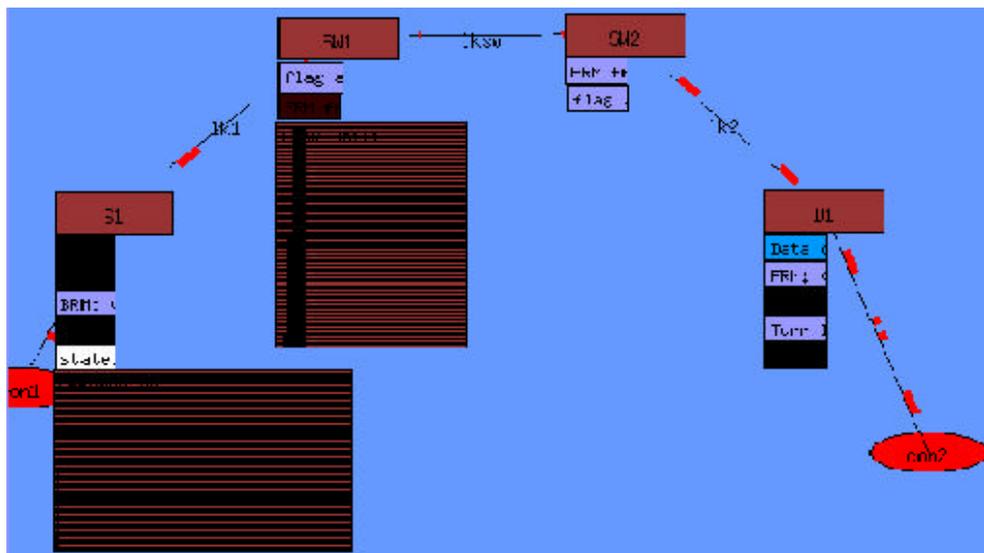


Figure 2. Network Topology

- High Throughput
- Bounded Oscillations
- Minimum Delay
- Congestion Avoidance
- Using Measured Rather Than Declared Overload
- The Scheme works for Bursty Traffic
- Minimal number of parameter
- Parameter Insensitivity
- Ease of Setting Parameters
- Order 1 Operation
- Bipolar Feedback
- Using input rates rather than queue length as the load measure
- Fairness is achieved without any fair queueing.
- Feedback is Related to Control.

2 84715 139.5
 2 104715 139.5
 2 124715 139.5
 2 144715 139.5

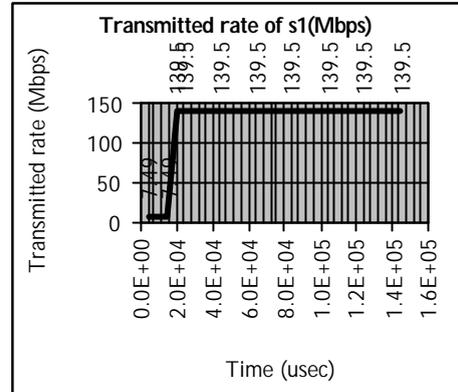


Figure 3

4. Some numerical results.

The followings are the typical example being produced by the simulator.

Network topology are composed of the two switches (SW1, SW2), one source (S1) and one destination (D1). They are illustrated in figure 2:

1 'SW1' 'Link Utilization of lksw' (figure 4)

1 10000 5.47097
 1 20000 5.47097
 1 30000 68.3871
 1 40000 90.271
 1 50000 90.271
 1 60000 90.271
 1 70000 90.271
 1 80000 87.5355
 1 90000 90.271
 1 100000 90.271

Impacts of Averaging Interval (AI) parameter:

- SW1: AI=100 (μsec); TU=0.9; Half-width-of-TUB=0.1
- SW2: AI=100 (μsec); TU=0.9; Half-width-of-TUB=0.1

2 'S1' 'Transmitted cell rate (Mbps)' (figure 3)

2 4715 7.49
 2 14715 7.49
 2 20752 139.5
 2 24715 139.5
 2 44715 139.5
 2 54715 139.5

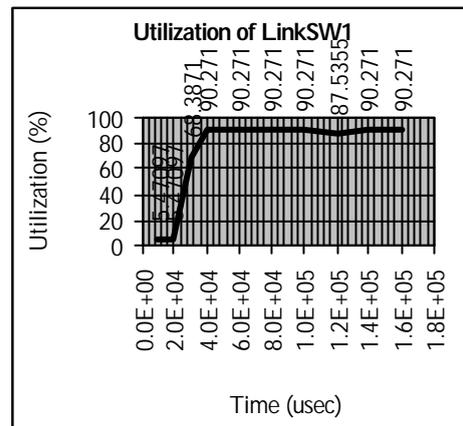


Figure 4

- SW1: AI=300 (μsec); TU=0.9;
Half-width-of-TUB=0.1
 - SW2: AI=300 (μsec); TU=0.9;
Half-width-of-TUB=0.1
- # 1 'S1' 'Transmitted cell rate (Mbps)' (figure 5)
- 1 4715 7.49
 1 34715 7.49
 1 43405 139.5
 1 64715 139.5
 1 94715 139.5
 1 124715 139.5
 1 154715 139.5
 1 184715 139.5
 1 214715 139.5
 1 244715 139.5

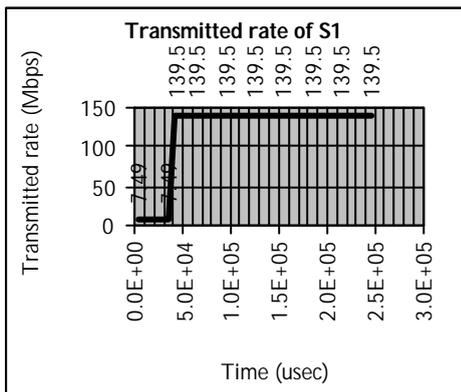


Figure 5

- # 2 'SW1' 'Link Utilization of lksw' (figure 6)
- 2 30000 5.47097
 2 60000 46.5032
 2 90000 90.271
 2 120000 90.271

- 2 150000 90.271
 2 180000 89.3591
 2 210000 90.271
 2 240000 90.271
 2 270000 89.3591
 2 300000 90.271

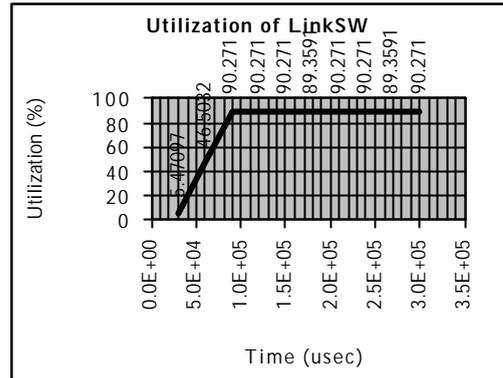


Figure 6

- SW1: AI=500 (μsec); TU=0.9;
Half-width-of-TUB=0.1
- SW2: AI=500 (μsec); TU=0.9;
Half-width-of-TUB=0.1

- # 2 'S1' 'Transmitted cell rate (Mbps)' (figure 7)
- 2 4715 7.49
 2 54715 7.49
 2 60382 139.5
 2 104715 139.5
 2 154715 139.5
 2 204715 139.5
 2 254715 139.5
 2 304715 139.5

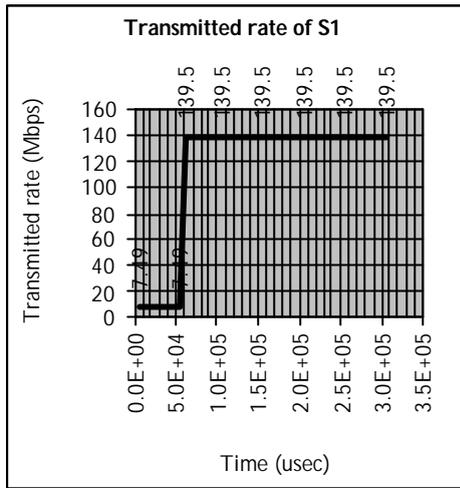


Figure 7

1 'SW1' 'Link Utilization of lksw' (figure 8)

- 1 50000 4.92387
- 1 100000 69.4813
- 1 150000 90.271
- 1 200000 89.7239
- 1 250000 90.271
- 1 300000 89.7239

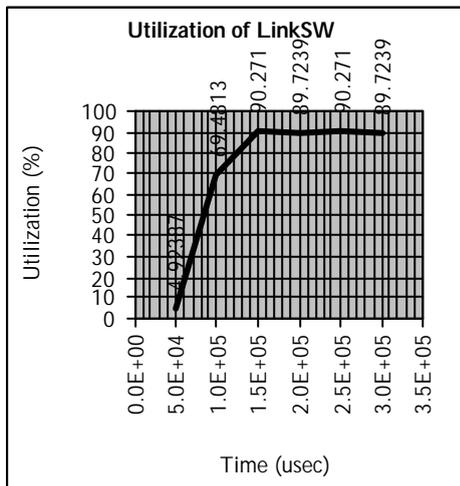


Figure 8

- + With AI=100(μsec), and when t=20752(μsec) source speed is

established and when t=40000(μsec) LinkSW link will reach its required link utilization.

- + With AI=300(μsec), and when t=43405(μsec) source speed is established and when t=90000(μsec) LinkSW link will reach its required link utilization.
- + With AI=500(μsec), and when t=60382(μsec) source speed is established and when t=150000(μsec) LinkSW link will reach its required link utilization.

In fact, when AI is enough small the switch status and control cell cycle will be more carried out, and so network status will be more frequently updated. However, in generality, the more small AI values are not good because it is easy to cause the congestion.

Impacts of Target Utilization (TU) parameter:

- SW1: AI=300 (μsec); TU=0.8; Half-width-of-TUB=0.1
- SW1: AI=500 (μsec); TU=0.9; Half-width-of-TUB=0.1

1 'S1' 'Transmitted cell rate (Mbps)' (figure 9)

- 1 4715 7.49
- 1 34715 7.49
- 1 43405 124
- 1 64715 124
- 1 94715 124
- 1 124715 124
- 1 154715 124
- 1 184715 124
- 1 214715 124
- 1 244715 124

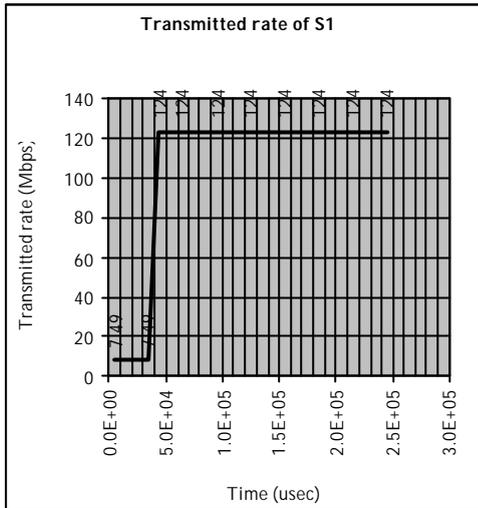


Figure 9

2 'SW1' 'Link Utilization of lksw' (figure 10)

2 30000 5.47097
 2 60000 41.9441
 2 90000 80.2409
 2 120000 79.329
 2 150000 80.2409
 2 180000 80.2409
 2 210000 80.2409
 2 240000 79.329
 2 270000 80.2409
 2 300000 80.2409

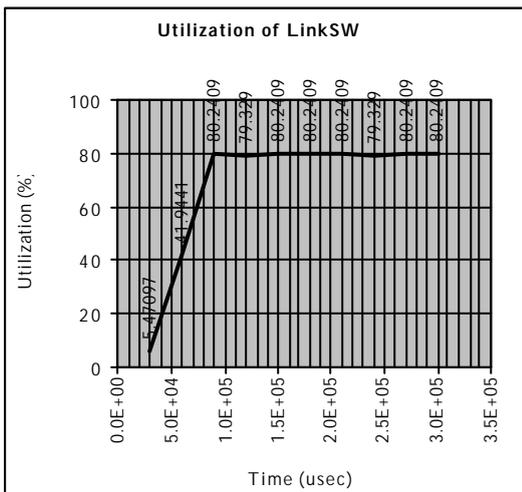


Figure 10

Although virtual channels going through switch structure require the different target utilization links, algorithm still supports to system running well as the smallest TU.

5. Conclusions.

The ATM network simulator has been applied to provide a means for our researchers, lecturers and senior students to analyze the behavior of ATM networks without the expense of building a real network. The simulator is tool that gives the user an interactive modeling environment with a graphical user interface. With this tool the user may create different network topologies, control component parameters, measure network activity, and log data from simulation runs. It is composed of two major tools: one is engaged in designing the network topology and, another one is a protocol analyzing tool. We use that tool to design the different network architectures and analyze the congestion avoidance solutions with the heterogeneous traffic sources. Particularly, we simulate the OSU congestion avoidance solution with the ABR quality level traffic sources on several network topologies. The numerical and graphical results help users easy making the performance evaluations. Many years since, we have applied that achievement to effectively support the research and training on networking subject at PTIT (Vietnam Posts and Telecoms Institute of Technology).

References

- [1] Anna Charny, David Clark, Rai Jain, *Congestion Control With Explicit Rate Indication*, AF-TM 94-0692, July 1994.
- [2] Anna Charny, *An Algorithm for Rate Allocation in a Cell-Switching Network with Feedback*, MIT TR-601, May 1994.
- [3] Rai Jain, Shiv Kalyanaraman and Ram Viswanathan, *The OSU Scheme for Congestion Avoidance in ATM Networks using Explicit Rate Indication*, OSU-CISRC-1/96-TR02.
- [4] Nada Golmie, Yves Saintillan, Alfred Koenig, David Su, *The NIST ATM/HFC network Simulator version 3*, March 1998.