# CHAPTER 5

# Simulation Results and Performance Analysis

## 5.1 Introduction

The previous chapter has presented the design of the Fuzzy Logic Traffic Controller and its related components. The algorithm and detailed operation of the Fuzzy Policer (FP) and Fuzzy Congestion Controller (FCC) has also been discussed. In this chapter, simulation experiments are presented to analyze the performance of the FP and FCC. Their performance is then compared to existing algorithms.

The first section looks at the simulation model and the method being employed in the simulation program. Next, the performance of the FP and FCC is compared to existing algorithms through simulations. Finally, a discussion is done at the end of the chapter based on the simulation results.

## 5.2 Simulation Model

The simulation model being employed is designed to simulate the functions performed by the usage parameter control on cells over VCCs passing through a network node of an ATM network. The FP and FCC schemes are included into the simulation model to measure their performance in policing cell rates and preventing congestion. Existing schemes like the leaky bucket algorithm are also included. The model simulates the traffic consisting of ATM cells flowing within the VCCs. Appropriate traffic source model is used to generate different types of traffic streams within the VCCs.

### 5.2.1 Objective

The objective of the simulation is to analyze the performance of the FP and FCC in policing and controlling traffic flow within VCCs passing through a network node. Comparison is made between existing schemes and the proposed FP and FCC to evaluate any performance gain. Figure 5.1 below shows the network topology used for all the simulation experiments. This topology is intended to represent a typical ATM network whereby a number of users are connected to an outgoing link via a network node (switch).

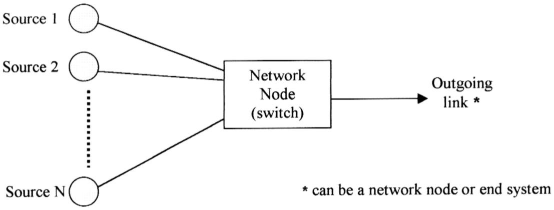


Figure 5.1 Simulation Network Topology

The outgoing link is a shared link to another network node or an end system (server). VCCs carrying data traffic goes through the network node to the outgoing link. The simulation program will generate traffic streams for each source and feed them through the network node via the VCCs. The purpose of the simulation is to monitor the traffic flow through the network node. In addition, the buffer state of the network node is also being monitored. From the characteristics of incoming cells such as their arrival time and queue length of the buffer, system performance measures can be derived and deduced. Thus, the performance of the FP and FCC can be analyzed through the simulation by varying the characteristics of the incoming traffic and buffer size.

### 5.2.2 Input

Details of all the virtual connections that go through the network node serves as input to the simulation program. Information such as the traffic type, traffic attributes like mean rate and mean burst size are given to describe each virtual connection. This information will be used by the respective traffic control functions to perform its task. Also the traffic source model will use these information to generate the type of traffic coming from the source.

### 5.2.3 Traffic Source Model

The traffic model, which is widely used for the characterization of ATM sources, is the on-off source model. This model has been successfully used to realistically model packetized speech, still picture and interactive data services [3, 22]. Using the on-off model, the ATM cell stream from a single source is modeled as a sequence of alternating burst periods and silence periods. During the "on" state, which is the burst state, the source will generate a stream of cells that are equally spaced at its peak rate. When it is in the "off" state, during the silence period, it generates no cells at all.

Two traffic generating models can be used to generate the traffic stream. In the first, both the burst and silence periods are exponentially distributed. In the second model, the number of cells generated during a burst follows the geometrical distribution, while the silence period is exponentially distributed.

Figure 5.2 and Figure 5.3 below illustrates the on-off source model and the characteristics of its traffic model:

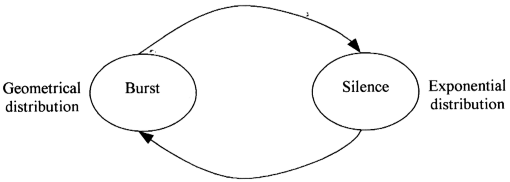


Figure 5.2 On-off Source Model

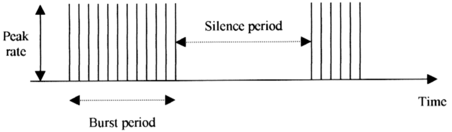


Figure 5.3 Characteristics On-off Source Model

By using an on-off burst silence model, the peak rate, mean burst size and the silence period are sufficient to completely describe the traffic statistics behavior of the source [3, 10, 22, 23]. The peak rate is the cell rate during a burst. Therefore, the cell interarrival time during a burst can be derived from the peak rate. The mean burst size is the average number of cells in a single burst.

Some important parameters to describe an on-off source are:

1) peak cell rate $p$
2) mean burst period $\lambda$
3) mean silence period $\mu$
4) burstiness $\beta = \frac{(\lambda + \mu)}{\lambda}$
5) mean cell rate $m = \frac{p * \lambda}{(\lambda + \mu)}$
6) mean burst size $b = (p * \lambda)$

For a review of other traffic models, please refer to [30].

### 5.2.4 Random Number Generator

The on-off source model used in the simulation has a geometrically distributed burst size and exponentially distributed silence period. In order to generate traffic following this model, the simulation program is required to generate random numbers having the above distributions. Therefore two random number generators are implemented in the simulation program for this purpose.

a) Random variable with exponential distribution

A few steps are involved to simulate random variable with exponential distribution. The first step is to generate a random variable U, with uniform distribution. This random variable can be generated by calling the random generator function rand() or random() in Borland C. This random variable is then converted to an exponential random variable X by

$$X = -a \ln (U)$$ a is the mean for the exponential distribution

This is called the Inverse Function Method [31].
Note: X is a continuous variable

b) Random number with geometric distribution

By using the Inverse Function Method:
The first step is to generate a random variable U, with uniform distribution. This random variable can be generated by calling the random generator function rand() or random() in Borland C. This random variable is then converted to a geometric random variable X by

$$X = \ln (U) / \ln (1 - p)$$ p is 1/m;
m is the mean for the geometric distribution

Note: X is a discrete variable

### 5.2.5 Data Structure

The details for each connection is represented by a data structure of the following form:

```
typedef struct source {
        float time_to_next;             /*time to next cell output*/
        float agreed_cell_rate;         /*agreed original cell output rate*/
        float on_off_time;              /* on/off time */
        char on_off;                    /* on or off indicator */
        float min_inter_time;           /*minimum interarrival time at peak rate*/
        int burst_size;                 /*average number of cells in single burst*/
 } SOURCE;
```

This data structure will be used in the simulation of FCC.

## 5.3 Simulation Results for the FP

The FP's goal is to simultaneously police mean rate and reject bursts. In this simulation, the performance of the FP is compared to the Leaky Bucket Algorithm. Both the FP and leaky bucket algorithm are performed on a per VCC basis. The result is based on the policing functions performed by both schemes on the same VCC. Identical sets of traffic pattern are fed to both the FP and the leaky bucket algorithm. The system performance is then measured in terms of cell loss ratio for the traffic in the VCC.

The optimized value for the parameters characterizing the fuzzy logic control is found to be:

For $A_1$,

$C_e = 1.0$, $C_W = 0.1$

$SC_e = 1.1$, $SC_w = 0.1$

$V_e = 1.2$, $V_w = 0.1$, $V_{max} = 3.0$

For $A_2$,

$C_e = 1.0$, $C_W = 0.15$

$SC_e = 1.15$, $SC_w = 0.15$

$V_e = 1.3$, $V_w = 0.15$, $V_{max} = 3.0$

For y,

$N_e = -0.01$, $N_w = 0.02$, $y_N = -0.2$

$P_e = 0.01$, $P_w = 0.02$, $y_P = 0.2$

For c,

$D_e = 0$, $BPD_e = 0.7$, $P_e = 1$

These parameter values can be adjusted according to preferences. For example, the parameter values can be set to be more sensitive to violation of mean rate and give a slight tolerance on the violation of burst size. Therefore, the FP is very flexible in handling different kind of scenarios.

The simulation is based on bursty traffic sources with geometrically distributed burst size and exponentially distributed silence period. The FP is tested in parallel with the leaky bucket algorithm. Two leaky buckets are used for comparison, both dimensioned to monitor mean cell rate:

1) leaky rate close to mean rate with a very large bucket (Leaky Bucket 1)
2) leaky rate much greater than mean rate with a very small bucket size (Leaky Bucket 2)

Two traffic types are considered: packetized voice and still images. The traffic characteristics of these traffic are based on data from [3], [22] and are shown in table 5.1 below:

| Source Parameter | Packetized Voice | Still Images |
|---|---|---|
| Peak rate, $p$ | 32 Kbps | 2 Mbps |
| Mean rate, $m$ | 11.2 Kbps | 87 Kbps |
| Burst Size, $b$ | 26 cells | 2358 cells |
| Silence period, $\mu$ | 0.65 s | 11s |
| Burstiness, $\beta$ | 2.85 | 23 |

Table 5.1 Traffic Characteristics of Packetized Voice and still images

Testing is performed on both violation of mean bit rate and violation of mean burst size. The simulation results obtained are as follows:

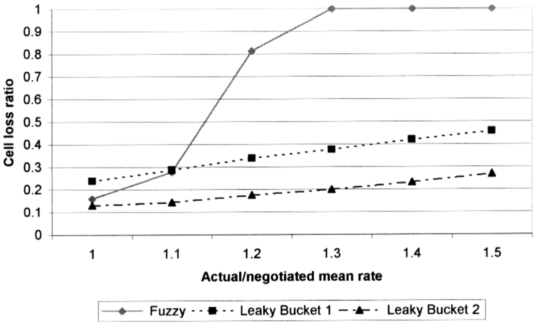


Figure 5.4 Violation of Mean Bit Rate for Packetized Voice

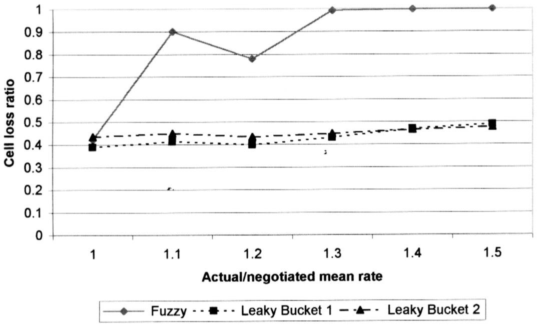


Figure 5.5 Violation of Mean Bit Rate for Still Images

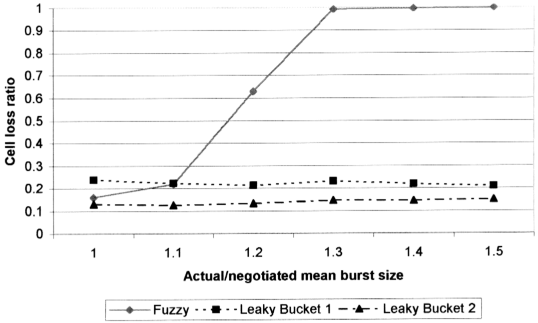


Figure 5.6 Violation of Mean Burst Size for Packetized Voice

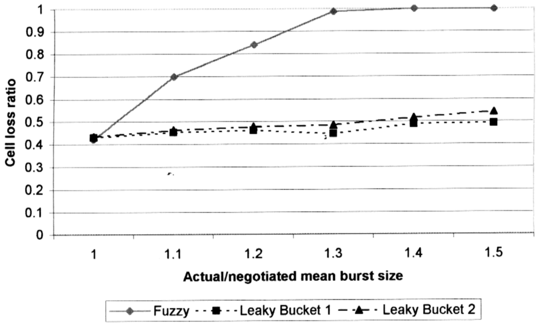


Figure 5.7 Violation of Mean Burst Size for Still Images

From the simulation results, it can be seen that the proposed FP outperforms the leaky bucket algorithm in both violation of mean rate and mean burst size. The FP performs extremely well on the policing of mean rate when compared to the leaky bucket algorithm. When the violation level is increasing, the FP is able to detect the situation and take appropriate action to discard and shut off sources that violate the negotiated values.

In the policing of burst size, the FP once again outperforms the leaky bucket algorithm. The leaky bucket algorithm is found to be less effective in policing burst size as compare to the FP. The FP is able to detect violation of burst size very efficiently thus dropping cells when necessary to ensure fairness and protect the network from severe congestion.

When comparing the performance of Leaky Bucket 1 and Leaky Bucket 2, two scenarios are observed. Firstly, results showed that Leaky Bucket 1 is more efficient in policing mean rate and burst size than Leaky Bucket 2 for packetized voice traffic. However, for still images, Leaky Bucket 1 is found to be slightly inferior to Leaky Bucket 2. The reason is due to the traffic pattern of incoming cells. Still images are more bursty in nature than packetized voice. Therefore, Leaky Bucket 1, which is more sensitive to violation of mean rate but inefficient in policing long bursts suffers in terms of performance when the traffic is more bursty (i.e. still images). On the other hand, Leaky Bucket 2 is dimensioned to be more effective in policing long bursts but less sensitive to violation of mean rate. Thus, Leaky Bucket 2 performed slightly better in policing still images.

In Figure 5.4 and Figure 5.6, it is found that Leaky Bucket 1 seemed to perform better within the range of 1 to 1.1 (actual / negotiated values) when compared to the FP. Leaky Bucket 1 is able to detect slight violation (between 1 to 1.1) and begin discarding cells. However, the objective of the FP is to tolerate a slight violation of the negotiated values (e.g. mean rate) when the other parameters (e.g. burst size) are below negotiated values. The purpose is to minimize reduction in throughput when cells are discarded. Therefore, the performance of FP within this region (1 to 1.1) is more desirable. Furthermore, the FP parameters can be adjusted to enforce a stronger action when the violation level is within this range.

Simulation results have demonstrated that the proposed FP can outperform the leaky bucket algorithm by substantial margin and the FP is very suitable for policing bursty traffic.

## 5.4 Simulation Results for the FCC

The Fuzzy Congestion Controller (FCC) is a Fuzzy Logic Controller (FLC) that can prevent or relieve network congestion. In this simulation, performance of the FCC is compared to the one that does not utilize fuzzy logic control. By comparing the cell loss ratio for both of them, the performance of the FCC can be analyzed.

The buffer size has been set to a capacity of 10 cells. The optimized parameter values for the fuzzy logic control is found to be:

$E_e = 4, \ E_w = 4$

$F_e = 8, \ F_w = 4, \ K_i = 10$

$N_e = -4, \ N_w = 7$

$P_e = 3, \ P_w = 7$

$DC_e = -0.2, \ NC_e = 0, \ I_e = 0.2$

The FCC, after evaluating the current buffer utilization, will send a rate control signal $y$, to notify the source to adjust its transmission/cell rate. The source will update its current rate as follows:

$$\text{new\_rate} = \text{current\_rate}*(1 - y)$$ (new_rate and current_rate is expressed in the form of mean interarrival time)

When the system is relieved from congestion, the transmission/cell rate will be restored to its original value to improve overall throughput and performance of the system.

The simulation is based on bursty traffic sources with geometrically distributed burst size and exponentially distributed silence duration. By using this model, the peak rate, mean burst size and the mean silence duration is sufficient to completely describe the traffic statistics behavior of the source. The FCC is put into test under different traffic environment. To accomplish this, traffic sources with different load conditions are used in the simulation. This is done by varying the above traffic parameters, i.e. peak rate, mean burst size and the mean silence duration.

Note: Cell interarrival time during bursts can be derived from the peak rate.

The parameters for the output link (network node) in the simulation are:

Mean Interarrival Time = 1 ms
Mean Burst size = 100 cells
Mean Silence duration = 200 ms

The simulation results are presented below:

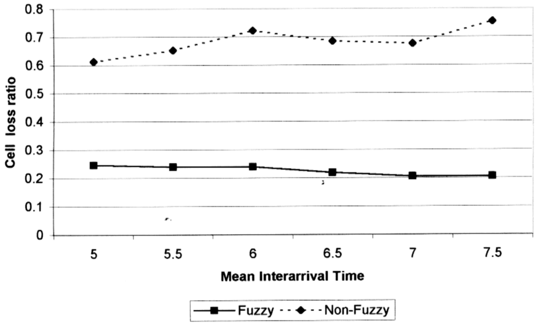


Figure 5.8 Variation of Mean Interarrival Time

Mean Burst size = 50 cells, Mean Silence duration = 500 ms

Number of connections = 10

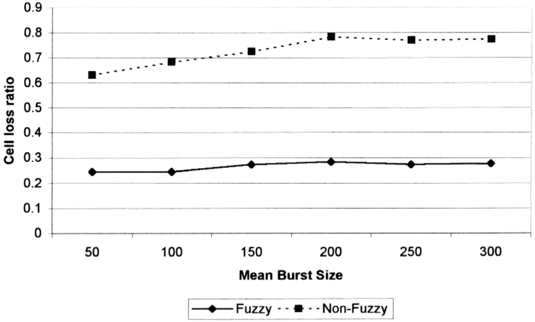


Figure 5.9 Variation of Mean Burst Size

Mean Interarrival Time = 5 ms, Mean Silence duration = 500 ms
Number of connections = 10

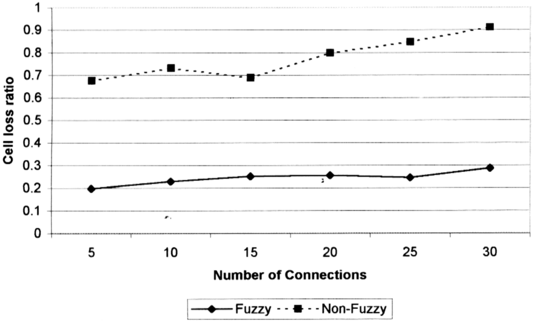


Figure 5.10 Variation of Number of Connections

Mean Interarrival Time = 5 ms, Mean Burst size = 50 cells
Mean Silence duration = 500 ms

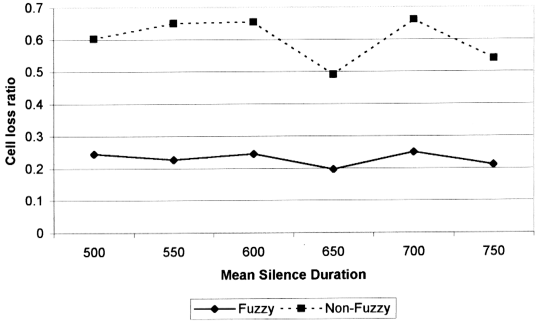


Figure 5.11 Variation of Mean Silence Duration

Mean Interarrival Time = 5 ms, Mean Burst size = 50 cells
Number of connections = 10

From the simulation results, it is evident that the proposed FCC significantly reduces congestion and reduces the cell loss ratio by substantial margin. The FCC provides significant reduction of cell loss due to congestion and buffer overflow under various traffic load environments. This is because the FCC will notify the source to adjust its transmission/cell rate according to the state of the system. If there is congestion, the source will need to reduce its current rate. However, when the system is relieved from congestion, then the transmission/cell rate will be restored to its original value. This is to improve overall throughput and performance of the system.

In Figure 5.8, the peak rate of the sources is varied to create different load conditions on the network node. Congestion occurs when the buffer at the network node overflows and cells are dropped. The cell loss ratio for the FCC scheme is compared to the one without any control. Results showed that the FCC performed very well in minimizing the occurrence of congestion at the network node. This is accomplished by giving feedback to the sources to adjust their transmission/cell rate. Thus, the FCC provides a significant reduction in cell loss ratio.

The average burst size for all connections (sources) are varied to test the performance of the FCC under different load environments. The results in Figure 5.9 once again showed the superiority of the FCC in minimizing the occurrences of congestion. The cell loss ratio for the FCC is much lower compared to the one without any fuzzy control. The number of cells lost due to buffer overflow is decreased sharply with the use of FCC.

Congestion sometimes occurs when the number of connections going through a network node increases. This may be due to new users entering the network to transmit data. Therefore, a simulation is performed whereby the number of connections is increased. The performance of the FCC is put into test under this environment. The results in Figure 5.10 clearly showed that the FCC achieved substantial reduction in cell loss ratio when compared to one without any control.

Finally, the mean silence duration for all the sources is varied to create different load environments on the network node. Once again, the cell loss ratio is used as a performance measure to determine the effectiveness of the FCC. Figure 5.11 illustrates the results obtained from the simulation. It has been found that the FCC reduces the number of cells lost due to congestion and buffer overflow, as shown by the low cell loss ratio obtained.