Impact of Priority-based ATM Switch Design on System Performance

Jianxun Ding
Dept of Computer Science
Texas A&M University
College Station, TX 77843, U.S.A.

Hong Jiang
Dept of Computer Science & Engineering
University of Nebraska
Lincoln, NE 68588, U.S.A.

Abstract

This paper concerns the impact of priority-based ATM switch design on system performance. The switch design consists of static buffer splitting, adaptive buffer splitting, as well as the priority-based selection strategies for solving cell contentions. A new analytical model is developed for the favorite load traffic pattern which can reflect some typical ATM applications and also include the uniform load as its special case. System performance comparisons are conducted for a number of different priority-based ATM switch designs.

1 Introduction

Asynchronous Transfer Mode (ATM) is the recommended transport technique for Broadband ISDN (BISDN) by CCITT [1]. Packets in the ATM transmission are small, fixed size information units called cells. Many ATM switch designs have been proposed to achieve high transmission capabilities. Among them, the self-routing multistage interconnection networks (MINs) are commonly regarded as one the effective network fabrics to build ATM switches in the high-speed transport systems supporting broadband services [1].

In this paper, we investigate the performance impact of a Banyan type MIN ATM switch with priority-based designs. In an $N \times N$ Banyan network, there are $\log_2 N$ stages, each stage having $N/\beta \times \beta$ switch elements (SEs). If there are more than one cells destined to the same outlet in an SE, a conflict occurs and only one cell can be selected to pass that outlet to the next stage. Usually, buffers are introduced to store the remaining cells. There are several ways to organize the buffers in an SE [2]. In the Conventional-buffered MINs (CMINs), every inlet of an SE has one FIFO buffer. Such buffer design has a drawback: When a conflict occurs, the unchosen cell at the head of a buffer will block the succeeding cells destined to other possible idle outlets. One natural way to solve this problem is to split every input buffer into several small ones, each small buffer corresponding to one outlet. Thus an unchosen cell will not block the succeeding cells destined to available outlets. The sizes of the split buffers can be fixed such that all the split buffers associated with an inlet have the same sizes; or they can be changed adaptively during the channel connection phase according to the traffic load. In other words, the buffer splitting need not be even: One split buffer can have larger size than others in the same inlet so that it can host more cells, avoiding possible cell losses. We call the MINs consisting of these two types of SEs the Split-buffered MINs (SMINs). Specifically, we call the first type of SMINs the -Statistically Split-buffered MINs (SSMINs) and the second type the Adaptively Split-buffered MINs (ASMINs). Fig. 1 shows an 8x8 Adaptively Split-buffered Banyan network — the Omega network.

In paper [3], we analyzed the SSMINs under the uniform input load. However, as mentioned above, there are numerous services in BISDN. The traffic loads in the BISDN ATM switches are highly nonuniform. Hence we should not limit our analyses under the uniform traffic pattern for the ATM switch design. On the other hand, performance analyses cannot exhaust all nonuniform traffic patterns. Consequently, more and more researchers are focusing on typical nonuniform traffic patterns that can reflect communication applications more realistically than the uniform traffic patterns [4, 5, 6].

This paper deals with the analysis of the priority-based SSMINs and ASMINs under the favorite load. The favorite load is a typical nonuniform traffic pattern and is more general than the uniform traffic pattern. As pointed by Bhuyan in [4], it is one of the common processor-memory communication patterns in the parallel processing environment. It also can reflect some current and future ISDN applications. The paper is organized as follows. Section 2 describes the features of SSMINs and ASMINs, the favorite load, and the priority-based selection strategies. Section 3 presents a new analytical model for various priority-based SSMIN and ASMIN switch designs. Section 3 discusses the analytical results and shows the interesting performance behavior of the priority-based SSMINs and ASMINs. Finally section 5 concludes the paper.
2 Preliminaries

Under the uniform traffic pattern, cells are directed uniformly over all of the network outlets. However, as Bhayan pointed out in [4], in an $N \times N$ multistage interconnection network, one network inlet is likely to send cells to one specific network outlet most of the time. If a network inlet IN, sends cells more often to the network outlet OUT, IN, will be called the favorite network inlet of OUT, OUT, the favorite network outlet of IN, and the cells from IN to OUT, the favorite cells. Assume that $\lambda$ is the probability that a network inlet generates favorite cells. The rest of the cells (non-favorite cells) are equally distributed to all other network outlets as the background traffic. We call this type of nonuniform traffic pattern as the favorite load.

The favorite load may appear in some BISDN applications when each network inlet sends a large amount of (image) data to its favorite network outlet and occasionally sends a small amount of (telephone) data to all other outlets. Note that in an $N \times N$ network, if cells from a network inlet IN, go to its favorite network outlet with the favorite rate $\lambda = 1/N$, the cells are uniformly distributed. So the uniform load is a special case of the favorite load.

It should be noted that the favorite rates $\lambda$'s in the load matrix are not necessarily being fixed in the diagonal positions. They can be in any other positions as long as no two $\lambda$'s are in the same row, i.e., any inlet cannot have two favorite outlets at the same time. Since the switch fabric we are discussing is permutationally symmetric, without loss of generality, we assume that the inlet IN,'s favorite outlet is OUT, in a split-buffered Omega network (see Fig. 1). It is easy to see that the favorite cells always pass through the straight connections exclusively. A non-favorite cell sometimes, but not always, passes through the straight connections. In an SE, a split buffer associated with the straight (cross) connection path is called the favorite (non-favorite) buffer. Any cell passing through an SE's favorite buffer is called the favorite request to that SE; other cells passing through non-favorite buffers are called non-favorite requests to that SE.

In addition to the above features, our analysis makes the following assumptions as in most of the conventional-buffered MIN analyses.

(a) The network is operated in a cycle-by-cycle mode.

(b) Cells are generated at each network inlet with the same probability. If in a cycle a cell cannot be accepted by a first stage SE because of the full buffer, it will be rejected. In the next cycle, the cell arrival probability and favorite rate will remain same.

(c) Cells generated in a cycle are independent of the cells generated in the previous cycles.

(d) There is no blocking at the outlets of the network.

3 Analytical Modeling

3.1 Denotations

In our analysis, we define the following variables.

- $b$ : the SE size. Note that our analysis is generalized to the split-buffered delta networks with $b \times b$ SEs.
- $n$ : the number of network stages.

Note that our analysis is generalized to the split-buffered SEs. Hence $L = b^N$.

- $B$ : the total buffer space in each SE inlet.

- $I_f$ or $In_f$ : the size of a split favorite or non-favorite buffer. Hence $L = I_f + (b - 1) \times In_f$.

- $l$ : the size of a split buffer in the SSMNs. Note in SSMNs, $l = In_f = L/b$.

- $p_f(k)$ or $p_n(k)$ : the probability that a favorite or a non-favorite buffer at the kth-stage has i cells at the beginning of a cycle. We define $z$ as $1 - x$ for any variable $x$, e.g., $p_f(k) + p_n(k) = 1$.

- $q(k)$ : the probability that a cell has left a $(k - 1)$th-stage SE and appeared on one of the $b$ inlets of a kth-stage SE during a cycle. Note that $q(1)$ represents the cell input rate to the network.

- $e(k)$ : the probability that a cell to a kth-stage SE inlet will go to its favorite buffer, provided that there is such a cell.

- $g_f(k)$ or $gn_f(k)$ : the fraction of $q(k)$ (a cell will go to a favorite or a non-favorite buffer) of a kth-stage SE. Note that $g_f(k) = q(k) \cdot e(k)$ and $q(k) = g_f(k) + (b - 1) \cdot gn_f(k)$.

- $e_f(k)$ : the fraction of $e(k)$ that a cell will go through all the straight connections (or all the favorite buffers) to the last stage outlet, i.e., to its favorite network outlet. Note that $e_f(1)$ is the fraction of real favorite cells issued from a network inlet to its favorite network outlet.

- $en_f(k)$ : the fraction of $e(k)$ that a cell will not go through all the straight connections (or all the favorite buffer queues) to a network outlet, but appearing to be a favorite request to a current kth-stage SE.

- $ed(k)$ or $ec(k)$ : the fraction of $q(k)$ that was due to a favorite or non-favorite requests at the previous $(k - 1)$th-stage SE. Note that $ed(k) + ec(k) = 1$.

- $ed(k)$ or $ec(k)$ : the fraction of $ed(k)$ or $ec(k)$ that a cell will go to a kth-stage SE's favorite buffer.

- $rf(k)$ or $rn_f(k)$ : the probability that a selected cell in a favorite or a non-favorite buffer at a kth-stage SE is selected among the contending cells to pass through the SE, given that there is a cell in that split buffer.

- $rf(k)$ or $rn_f(k)$ : the probability that a selected cell in a favorite or a non-favorite buffer at a kth-stage SE is able to move forward because the destined split buffer at stage $k + 1$ is not full.

- $S$ : normalized throughput: the average number of cells passed through the network per output link per cycle.

- $d$ : normalized delay: the average number of cycles taken for a cell to pass a single stage.

As we mentioned in section 2, if there are more than one cells destined to the same SE outlet at the same cycle, only one cell can be selected. Here we consider three selection policies: (1) the Equal Priority (EP) policy where cells in the favorite or non-favorite buffers receive equal priority; (2) the Priority to Favorite buffers (PF) policy where cells in the favorite buffer receive higher priority than the non-favorite buffers; and (3) the Priority to Non-Favorite buffers (PNF) policy where cells in the non-favorite buffers receive higher priority than the favorite buffer and conflicts
among non-favorite buffers are solved randomly. In the following subsections 3.2, 3.3, and 3.4, we will derive an analytical model for SSMDINs with these three selection policies. Then in subsection 3.5, we will revise it to get the analytical model for ASMDINs.

### 3.2 EP Policy

By examining the functional features of the switches, we were able to find some important quantitative relationships among various variables defined subsection 3.1. The following expressions represent these relationships. We assume that $1 \leq k \leq n$ unless otherwise specified.

$$
q(k + 1) = 1 - p_{f0}(k) \cdot pn_{f0}(k)^{k-1}. 
$$  
(1)

$$
ed(k + 1) = \frac{p_{f0}(k)}{b \cdot pn_{f0}(k)} q(k + 1) \cdot [1 - pn_{f0}(k)^{k}]. 
$$  
(2)

$$
ec(k + 1) = 1 - ed(k + 1). 
$$  
(3)

$$
ef(k + 1) = \frac{ef(k)}{ec(k)} \cdot ed(k + 1) + \frac{b}{N} \cdot ec(k + 1). 
$$  
(4)

$$
enf(k + 1) = \frac{N/b^{k+1} - 1}{N/b^{k} - 1} \cdot \frac{ef(k)}{ec(k)} \cdot ed(k + 1) + \left(1 - \frac{b}{N} \cdot ec(k + 1)\right). 
$$  
(5)

$$
e(k + 1) = ef(k + 1) + enf(k + 1). 
$$  
(6)

$$
edf(k + 1) = \frac{ef(k)}{e(k)} + \frac{N/b^{k+1} - 1}{N/b^{k} - 1} \cdot \frac{ef(k)}{ec(k)} \cdot ed(k + 1). 
$$  
(7)

$$
ecf(k + 1) = \frac{1}{b}. 
$$  
(8)

$$
ef(k + 1) = q(k + 1) \cdot ef(k + 1). 
$$  
(9)

$$
qnf(k + 1) = q(k + 1) \cdot qf(k + 1). 
$$  
(10)

$$
r(k) = \frac{1}{b \cdot pn_{f0}(k)} \cdot [1 - pn_{f0}(k)^{k}]. 
$$  
(11)

$$
rnf(k) = 1 - p_{f0}(k) \cdot pn_{f0}(k)^{k-1} - \frac{p_{f0}(k)}{b - 1} \cdot \frac{1}{pn_{f0}(k)^{k} - b \cdot pn_{f0}(k)^{k}}. 
$$  
(12)

$$
rf(k) = \frac{r(0) \cdot ed(k + 1) + p_{f0}(k)}{ed(k + 1) \cdot pn_{f0}(k) \cdot pn_{f0}(k)^{k + 1}}. 
$$  
(13)

$$
rnf(k) = rnf(k) \cdot ecf(k + 1) - \frac{rnf(k)}{ef(k + 1)} \cdot p_{f0}(k)^{k + 1}. 
$$  
(14)

$$
rnf(n) = rnf(n) \cdot qf(n). 
$$  
(15)

$$
pf_{i}(k) = \frac{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k)}{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k) \cdot p_{f0}(k)}. 
$$  
(16)

$$
pf_{i}(k) = \frac{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k)}{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k) \cdot p_{f0}(k)}. 
$$  
(17)

$$
pf_{0}(k) = \frac{1}{1 + \sum_{i=0}^{k} pf_{i}(k) / p_{f0}(k)}. 
$$  
(18)

The pnf formulas for a non-favorite buffer are in the similar format. Given the input load $q(1)$ and the favorite rate $ef(1)$, we have:

$$
qnf(k + 1) = q(k + 1) \cdot qf(k + 1). 
$$  
(19)

$$
rnf(k) = \frac{rnf(k)}{ef(k + 1)} \cdot p_{f0}(k)^{k + 1}. 
$$  
(20)

$$
pf_{i}(k) = \frac{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k)}{qf(k) \cdot r_{f}^{i - 1} \cdot p_{f0}(k) \cdot p_{f0}(k) \cdot p_{f0}(k)}. 
$$  
(21)

$$
pf_{0}(k) = \frac{1}{1 + \sum_{i=0}^{k} pf_{i}(k) / p_{f0}(k)}. 
$$  
(22)

With the steady input load $q(1)$ and $ef(1)$, we can solve the above equations iteratively and get the steady-state values of $rf(k)$, $rnf(k)$, $pf_{i}(k)$, and $p_{f0}(k)$, where $1 \leq k \leq n$ and $0 \leq i \leq l$. Then the performance figures $S$ and $d$ can be calculated as follows.

$$
S = 1 - p_{f0}(n) \cdot pn_{f0}(n)^{n-1} 
$$  
(23)

$$
d = \frac{1}{b} \cdot \frac{1}{n} \sum_{k=1}^{n} \frac{1}{Rf(k)} + \frac{b - 1}{n} \sum_{k=1}^{n} \frac{1}{Rnf(k)}. 
$$  
(24)

$$
Rf(k) = rf(k) \cdot \frac{[pf_{i}(k) / p_{f0}(k)]}{i}. 
$$  
(25)

$$
Rnf(k) = rnf(k) \cdot \frac{[pnf_{i}(k) / pnf_{0}(k)]}{i}. 
$$  
(26)

It should be noted that if we let $c(1) = 1/N$, then $c(1) = 1/b$. The uniform traffic pattern becomes a special case of the favorite load. This is validated by using these two values as the inputs to the above analytical model for various sized SSMDINs. All the results from the above favorite load model are exactly the same as those obtained from the uniform load model presented in [3].

### 3.3 PF Policy

Under the PF policy, the routing logic gives the favorite buffers higher priority than the non-favorite buffers. A cell in a favorite buffer is always selected whenever there is a conflict with the cells in the other $b - 1$ dedicated non-favorite buffers. Hence, the probability that a cell is ready to a $(k + 1)$-th stage SE’s inlet due to a favorite request is equal to the probability that the preceding stage SE’s favorite buffer is not empty, regardless of the statuses of the other non-favorite buffers. Therefore, the equation (2) about $ed(k + 1)$ should be changed to:

$$
q(k + 1) \cdot ed(k + 1) = p_{f0}(k). 
$$  
(27)

Hence,

$$
ed(k + 1) = \frac{p_{f0}(k)}{q(k + 1)}. 
$$  
(28)

Similarly, under the PF policy, a cell in a favorite buffer need not to compete with cells in the other non-favorite buffers in order to be selected. However, if the favorite buffer is empty, a cell in a non-favorite buffer must compete with cells in other $j$ ($0 \leq j \leq b - 2$) non-favorite buffers in order to be selected. So the equations (11) and (12) of $rf(k)$ and $rnf(k)$ should be changed to:

$$
rf(k) = 1.0. 
$$  
(29)

$$
rnf(k) = \frac{p_{f0}(k)}{(b - 1) \cdot pn_{f0}(k)} \cdot [1 - pn_{f0}(k)^{k-1}]. 
$$  
(30)
3.4 PNF Policy

Under the PNF policy, the routing logic gives non-favorite buffers higher priority than favorite buffers. A cell in a favorite buffer cannot be selected unless all the \( b - 1 \) non-favorite buffers dedicated to the same SE outlet are empty. Conflicts among cells in the non-favorite buffers are solved randomly. Hence, the probability that a cell is ready to a \((k+1)\)th-stage SE's inlet due to a favorite request is equal to the probability that the preceding stage SE's favorite buffer is not empty AND all its \( b - 1 \) non-favorite buffers are empty. Therefore, the equations (2) about \( ed(k + 1) \) should be changed to:

\[
q(k + 1) \cdot ed(k + 1) = \frac{p_{f_0}(k) \cdot pn_{f_0}(k)^{b-1}}{q(k + 1)}. \tag{32}
\]

Hence,

\[
ed(k + 1) = \frac{p_{f_0}(k) \cdot pn_{f_0}(k)^{b-1}}{q(k + 1)}. \tag{32}
\]

Similarly, under the PNF policy, a cell in a favorite buffer cannot be selected unless the corresponding \( b - 1 \) non-favorite buffers are all empty. A cell in a non-favorite buffer has higher priority to be selected than a cell in the favorite buffer; but it must compete with cells in other j (\( 0 \leq j \leq b - 2 \)) non-favorite buffers. So the equations (11) and (12) of \( r_f(k) \) and \( r_{nf}(k) \) should be changed to

\[
r_f(k) = \frac{pn_{f_0}(k)^{b-1}}{k - 1}, \tag{33}
\]

\[
r_{nf}(k) = \frac{1}{(k - 1)pn_{f_0}(k)} \left[ 1 - pn_{f_0}(k)^{b-1} \right]. \tag{34}
\]

3.5 Adaptive Splitting

One may naturally consider that under the favorite load, buffers should, instead of being equally split, be divided according to the ratio between the favorite cells and the non-favorite cells. We call this strategy adaptive splitting. Adaptive splitting can be either fixed for some specific applications, or dynamic for general applications. Tamir and Frazer reported an implementation of dynamic splitting in the UCLA ComCBB chips [2]. In their implementation, they allowed packets to have varied lengths and used virtual cut-through technique to further increase the switch performance. The implementation and hardware cost issues are beyond the scope of this paper. Here we restrict our discussions on the analytical modeling for ATM packet (cells with fixed length) switching strategy. Our main purpose is to derive an analytical model to evaluate the performance behavior of the adaptive splitting in combination with the priority-based selection policies.

Recall that \( U(L) \geq 3; L \leq 2 \) is meaningless for the adaptive splitting) represents the total number of buffer units associated with an SE's inlet. \( Lf \) represents the size of a favorite buffer, and \( Ln \) represents the size of a non-favorite buffer. There are many ways for the adaptive splitting depending on how to select the adaptive coefficient. We give one simple example here to show how to apply the analytical model. Let \( \delta \) be the adaptive coefficient.

\[
\delta = \frac{e(1)}{(1 - e(1))(b - 1)}, \tag{35}
\]

set \( \frac{L_f}{Ln} = C \cdot \delta, \) or \( \frac{L_f}{Ln} = \frac{C}{\delta}, \) and, \( L = Lf + (b - 1) \times Ln, \) \( (36) \)

where \( Lf \) and \( Ln \) must be integers greater or equal to 1; \( C \)

\[
\frac{\text{Normalized Throughput}}{256 \times 256 \text{ networks with } 2 \times 2 \text{ Sfs}}
\]

\[
\begin{array}{c|c|c|c|c}
\hline
\text{Input Rate } q(1) & 0.0 & 0.2 & 0.4 & 0.6 & 0.8 & 1.0 \\
\hline
\text{Throughput} & 0.8 & 0.6 & 0.4 & 0.2 & 0.0 & 0.0 \\
\hline
\end{array}
\]

Figure 2: Impact of EP Policy on System Throughput in SSIM and ASMIN

Our analytical model is validated through extensive simulations with confidence intervals of 95%. The analytical results are within 5% of the simulation results on the average and within 9% in the worst case scenario. Note that most conventional-buffered models have the low accuracy under the heavy input load [7]. One of the major reasons for this inaccuracy stems from the re-distribution assumption — a blocked cell in an SE is uniformly re-distributed in the next cycle. In a split-buffered MIN, a blocked cell will stay in the "dedicated" split buffer and compete for the same outlet again in the next cycle. Hence this re-distribution assumption is no longer used in our model.

Our analytical model is validated through extensive simulations with confidence intervals of 95%. The analytical results are within 5% of the simulation results on the average and within 9% in the worst case scenario. Note that most conventional-buffered models have the low accuracy under the heavy input load [7]. One of the major reasons for this inaccuracy stems from the re-distribution assumption — a blocked cell in an SE is uniformly re-distributed in the next cycle. In a split-buffered MIN, a blocked cell will stay in the "dedicated" split buffer and compete for the same outlet again in the next cycle. Hence this re-distribution assumption is no longer used in our model.

Fig. 2 compares the normalized throughput versus input rate for 256 x 256 sized SSIM and ASMIN with the EP strategy. The favorite rate \( e(1) = 0.75 \) and the buffer size of an SE inlet is 4. We use \( 2 + 2 \) to represent the evenly split-buffered SSIM with each split buffer size \( l = 2 \). The figure shows that ASMIN with the 3+1 splitting performs better than the SSIM with the 2+2 splitting. And the ASMIN with the 1+3 splitting has the lowest normalized throughput among the three splitting approaches. The results conform the intuition. The 3+1 splitting fits the favorite input load better than the 2+2 splitting because it can host more cells in the network. On the other hand, the 1+3 splitting is totally unsuitable for the favorite input load and thus has the lowest throughput.

As for the normalized delay, Fig. 3 and Fig. 4 show
again that EP strategy always gives moderate performance between the two priority-based strategies, regardless how the buffers are split. The PF gives very small delay for the favorite cells in the 3 + 1 splitting but causes extremely large delays for the non-favorite cells. For the 1 + 3 splitting, because of the limited buffer space for the majority cells, a lot of cells have to be waiting outside of the network. So the normalized delay within the network seems very small. The interesting result is that the PNF strategy does not cause extremely long delays compared with the PF strategy in both 3 + 1 and 1 + 3 cases. In fact, some applications needs this strategy: The small amount of exception/synchronization/telephone signals need to be handled in the higher priority than the large amount of normal/routine/image signals.

Overall, the PNF and EP strategy in both SSMNs and ASMINs does not cause extreme cell delays compared with the PF strategy; the PF strategy in ASMINs can be used in some extreme situations under which the favorite cells must have very short delays in the expense of the extremely long delays of non-favorite cells. The adaptive splitting with more buffers for favorite cells generally performs better than the evenly splitting and the adaptively splitting with more buffers for non-favorite cells. It should be noted that all the performance gains are in the expense of hardware complexities. So the problem of choosing which combination of cell selection and buffer splitting strategy should be based on the QoS requirements of the ATM cell transmissions. This paper only concerns the analytical modeling of the SSMNs and ASMINs. The hardware cost comparisons are beyond the scope of this paper and deserve another paper. Our ultimate goal is to develop a set of formal analytical models for various ATM switch designs and then use these models to find the most cost-effective switch fabrics.

5 Conclusions

In this paper, we analyzed the impact of priority-based ATM switch design on system performance. The switch designs involve static buffer splitting, adaptive buffer splitting, and the priority-based selection strategies for solving cell contentions. An analytical model has been developed for the favorite load traffic pattern. The favorite load is more general than the uniform load. It includes the uniform load as its special case and it can reflect some typical ATM applications. With the help of this analytical model, we have investigated system performance features such as network throughput and cell delay of favorite or non-favorite cells under different priority based selection strategies.

Acknowledgement

The authors wish to express their gratitude to Professor L. N. Bhuyan for his valuable comments and suggestions during the process of this study.

References