출구 버퍼모듈을 갖는 패킷교환식 상호 연결 망의 성능 분석

추 현 승† · 박 경 린††

요 약

패킷교환식 다단계 상호 연결 망은 디지털 교환시스템과 슈퍼 컴퓨터분야에서 널리 이용되어 왔다. 본 논문에서 교환 요소내부의 여러 개의 패킷이 한 네트워크 주기동안 그 주기의 대역폭을 충분히 사용함으로써 다음 단계의 교환 요소로 전송되어 질 수 있음을 보인다. 기존의 상호 연결 망에서는 보통 하나의 패킷 전송을 가정했다. 다중 패킷 전송방식이 도입된 다단계 상호 연결 망의 분석적 모델이 소개되고, 컴퓨터 시뮬레이션에 의하여 그 모델의 타당성을 입증한다. 단일 패킷 전송방식을 이용한 전형적인 다단계 상호 연결 망과의 비교는 실용적인 크기의 연결 망에 대하여 처리율이 약 30% 정도까지 증가함을 보여준다. 비슷한 결과가 지연시간에 있어서도 얻어진다. 네트워크 전송량이 불균등한 분포를 나타내는 경우의 성능 향상은 더욱 중요하다.

Performance Analysis of Packet Switched Interconnection Networks with Output Buffer Modules

Hyun Seung Choo† · Gyung-Leen Park††

ABSTRACT

Packet-switched multistage interconnection networks(MINs) have been widely used for digital switching systems and super computers. In this paper we show that multiple packets in a switching element can move to the succeeding switching element in one network cycle by fully utilizing the cycle bandwidth. Only one packet movement was usually assumed in typical MINs. We present an analytical model for the MINs with the multiple packet movement scheme, and validate it by computer simulation. Comparisons with the traditional MINs of single packet movement reveal that the throughput is increased up to about 30% for practical size MINs. Similar result was also obtained for delays. The performance increase is more significant when the network traffic is nonuniform.

1. Introduction

Packet-switched multistage interconnection networks(MINs) have been used for parallel computer systems[1]. Compared to single stage switching networks such as crossbar and triangular, they can provide a high processor-to-memory bandwidth[2,3] at a much lower hardware cost. MINs can also be fault tolerant by providing disjoint paths[4] between each source and destination pair. Due to these properties, MINs have also been recently attracted to the switching of digital data communication such as

* This work was supported by SEOK CHUN Research Fund, Sungkyunkwan University, 1998.
† 정 회 원 : 성균관대학교 전기전자 및 컴퓨터공학부 교수
†† 정 회 원 : 제주대학교 전산통계학과 교수
논문접수 : 1998년 7월 28일, 심사완료 : 1998년 9월 28일

Asynchronous Transfer Mode(ATM)[5] switches employed in Broadband Integrated Services Digital Network(ISDN).

Switching elements(SEs) of MINs are unbuffered[6-8] or buffered. For the multibuffered MINs where the buffers are used to increase the MIN performances and avoid the internal loss of the packets, the back pressure mechanism is usually assumed for the packet movement. Here a packet is transmitted to its destined buffer in the succeeding stage depending on the buffer space availability. In most earlier MIN designs, during one network cycle, the buffer availability information is first propagated backwards from the last stage to the first stage and then simultaneous packet movements between each adjacent stages take place. Such scheme is called *Big Clock Cycle(BCC)* scheme. Extensive studies have been done on analyzing the performance of MINs with the BCC scheme[9-14]. Ding and Bhuyan[15] introduced another scheme where the packet movements occur without the propagation of the buffer space availability. They showed that the performance can be significantly improved by employing the scheme called *Small Clock Cycle(SCC)* scheme, even though the actual improvement varies according to the size of the packet[16].

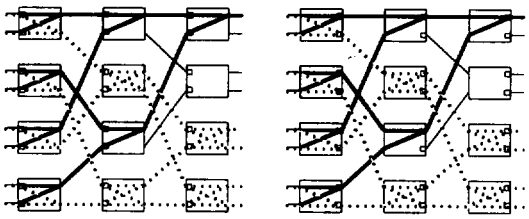
Another very important aspect related to the network operation is the length of a network clock cycle. As identified later, a network cycle must be long enough to guarantee that a head packet moves to the head position of the buffer in the succeeding SE. This, then, will be enough time for more than one packet to move in the pipeline fashion toward the succeeding stage. In this paper we show that the multiple packet movement mechanism can significantly enhance the network performance. We define the traditional scheme as single packet movement (SPM) scheme, while ours multiple packet movement (MPM) scheme, respectively. We develop an analytical model for MINs with the MPM scheme. Computer simulation validates the correctness of the model. Comparisons with the traditional MINs of the SPM scheme reveal that the performance enhancement is significant such that the throughput is increased up to about 30% for practical size MINs as the offered traffic load grows to 1. Similar result was also obtained for delays. More importantly, the performance enhancement gets more significant when nonuniform traffic is prevalent in the network.

The rest of the paper is organized as follows. In section 2, some important issues related to the packet movement in MINs and the analytical modeling of them are discussed. Section 3 presents the proposed model for multibuffered MINs with the MPM scheme. The model is verified through computer simulation, and compared with the SPM scheme in section 4. Finally, conclusions are made in section 5.

2. Operation Mechanism of MINs

In this section, the characteristics of the packet movement in MINs are investigated, which motivate the MPM scheme. As in [11], we refer a *buffer* module to a queue which has first-in first-out (FIFO) operation and a *buffer* to an individual buffer space. As mentioned earlier, buffers in each SE were introduced to packet switched MINs for achieving a practical level of network performance by holding the blocked packets. Through comprehensive simulation[17], though, it was identified that the buffer utilization decreases rapidly for the stages at the output side even under the uniform traffic condition. This is due to the contention between packets for the links or SEs, while the buffer module operation is FIFO. As a result, the buffer modules in the input side stages are usually saturated, while the output side stages hold few packets. This phenomenon gets more significant as the nonuniform traffic becomes more prevalent, and eventually tree saturation[18] occurs which seriously degrades the network performance. It is obvious that allowing the packets to move quickly is very important for achieving a high network performance.

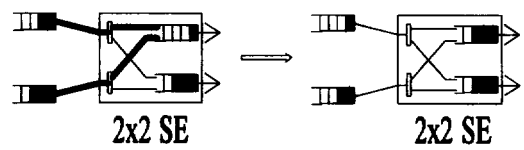
A number of approaches[19,20] have been proposed to minimize the performance degradation caused by tree saturation. These approaches, however, requires some substantial hardware and operational overhead to be implemented in the network. As shown here, though, the proposed MPM scheme can solve the problem without any significant overhead.



(Fig. 1) The comparison of input buffer and output buffer based MINs

For the multiple packet movement, first of all, buffer modules are put at output side of each SE. Figure 1 compares the effect of the position of the buffers- input or output side -with respect to the network performance. Here the topmost memory module is assumed to be the hot spot. The tree saturation condition is depicted using thick lines, while the paths for the traffic to the lower four memory modules are shown using the dotted lines. Notice from figure 1(*a*) that the traffic to the lower four memory modules compete with the nonuniform traffic for the buffers in the first stage which are suffering from tree saturation. Consequently the uniform traffic are unnecessarily blocked at the buffers, which results in the significantly degraded network performance. On the contrary, when the buffers are located at the output side of each SE as shown in figure 1(*b*), such unnecessary blockings are avoided. Notice that the uniform traffic move through the buffers not involved in the tree saturation. This property of disjoint paths for the traffics to the upper and lower half destinations will significantly enhance the performance compared to the input buffering scheme when nonuniform traffic is prevalent. This is verified later through computer simulation.

Another important advantage of output buffer is that each buffer module in the network is connected to two buffer modules in the preceding stage through disjoint paths. In the input buffer structure, however, the two preceding buffers share a common link. The property of disjoint paths in the output buffer case enables multiple packets to move in one network cycle. Note that, in one network cycle, a packet in the preceding stage moves to the receiving buffer module, and locates at the tail(next to the currently occupied buffer entry) of the buffer module. Let us denote m the size of a buffer module. Since a buffer module is basically a shift register, it takes actually $(m-b)\tau$ time units for a packet to move to the succeeding stage if there exist $b(0 \le b \le m)$ packets in the buffer module. Here τ is the time for shifting a packet one position. The worst case is when the receiving buffer module is empty, and then the length of a network cycle must be $m\tau$ to complete one packet movement between each pair of sending and receiving buffer module. As mentioned earlier, even with uniform traffic, the utilization of most buffer modules in the stages at the output side is much lower than 50%.



(Fig. 2) Two packet movements in one network cycle

For MINs with 2 × 2 SEs, for example, one buffer module can receive one packet from each of the two buffer modules in the preceding stage. This is shown in figure 2. If the size of the buffer modules is 4, then, the required period of a network cycle is 4τ. This is a sufficient time for receiving two packets. Figure 2 depicts a scenario where two packets destined to the same buffer module are allowed to move in one network cycle. The solid links represent this condition. Notice that the two

packets can move in the pipelined fashion since the paths are disjoint. In the traditional single packet movement(SPM) scheme, only one packet is accepted during one network cycle. The proposed multiple packet movement(MPM) scheme is expected to significantly enhance the performance of the network since the network congestion gets worse mostly due to the blocked packets which lost the contention. Other packets behind the blocked ones at the head of buffer modules are unnecessarily stalled. Using the MPM scheme, the chance of blocking will be reduced a lot. Several approaches[19,20] have been proposed to resolve this problem at the cost of some hardware and operational overhead. Our MPM scheme based on the output buffer structure is very simple which is important for implementing the scheme in practice.

3. Analytical Model with MPM scheme

3.1 Assumptions and Definitions

The assumptions required for the model are as follows. 1) 2×2 switching elements with the buffer modules of size m are used, where the buffer modules are located at the output port. The two buffer modules in a stage which are the two input sources of a buffer module in the succeeding stage are called a *contending pair*. 2) The traffic is uniform such that packets generated by inlets (sending/input nodes or processors) are distributed uniformly over all outlets(receiving/output nodes or memory modules). The nonuniform case is underway. The probability of a packet arrival to each network input is same for all network inputs. 3) Each packet has the same probability to win the contention, and the blocked packet is resubmitted to its original destination buffer module. 4) Output nodes are fast enough to accept a packet per network cycle from SEs at the last stage.

We assume that a network cycle consists of two phases as done in earlier designs. Here the sending buffer modules check the empty space availability of the receiving buffer modules in the first phase. Based on the information of the first phase, each sending buffer module sends a packet to the destination or enters into the blocked state in the second phase. Notice here that, as mentioned earlier, the SCC scheme is employed for packet movement where the empty buffer availability does not propagate backward[15,16].

In each network cycle the packets at the head of each buffer module(head packets) in a contending pair contend with each other if there exists only one empty buffer at the destined module. Based on the status of the packet at the head, the state of a buffer module can be defined as follows : 1) $State-0$: a buffer module is empty. 2) $State-n_i$ $(1 \le i \le m-1)$: a buffer module has i packets and the head packet moved into the current position in the previous network cycle. 3) $State-b_i$ $(1 \le i \le m)$: a buffer module has i packets and the head packet could not move forward due to the empty buffer unavailability in the previous network cycle, *i.e.* it has stayed at the position at least for one network cycle.

Note that there is no $State-n_m$. This can not occur in the SCC scheme, where a packet is accepted only when the buffer module is not full. Hence, $State-n_m$ does not exist.

The following variables are defined to develop our analytical model.

- $P_0(k, t)$: probability that a buffer module of SE(k) is empty at time t.
- $P_{n_i}(k, t)$: probability that a buffer module of SE(k) is in $State-n_i$ at time t, where $1 \le i \le m-1$
- $P_{b_i}(k, t)$: probability that a buffer module of SE(k) is in $State-b_i$ at time t, where $1 \le i \le m$
- $SP_n(k, t)$: $\sum_{i=1}^{m-1} P_{n_i}(k, t)$
- $SP_b(k, t)$: $\sum_{i=1}^{m} P_{b_i}(k, t)$
- $q(t)$: offered traffic load to the network inputs during $[t, t+1)$.

- $q_0(k,t)/q_1(k,t)/q_2(k,t)$: probability that no packet /one packet/two packets are ready to come to a buffer module of SE(k) during $[t, t+1)$.
- $T_0(k,t)/T_1(k,t)/T_2(k,t)$: probability that a buffer module of SE(k) receives no packet/one packet /two packets during $[t, t+1)$.
- $r_n(k,t)/r_b(k,t)$: probability that a normal/blocked packet in a buffer module of SE(k) can get to the destined buffer module in Stage $-(k+1)$ during $[t, t+1)$.

The state transition diagram for a buffer module can be developed using the variables, and it is shown in figure 3. Observe that, for example, $q_2 r_n$ is the transition probability from $State-n_1$ to $State-n_2$. Here q_2 represents the probability that two packets are ready to come while r_n does that for the packet leaving the buffer module. Clearly, $q_2 r_n$ results in the two packets in the buffer module with a normal head packet - $State-n_2$.

The following state equations are formulated from the diagram. The state of a buffer module at time $(t+1)$ is determined from the information of itself at time t. For simplifying the notation, (k,t) is omitted for all the variables.

$$P_0(k,t+1) = q_0(P_0 + r_n P_{n_1} + r_b P_{b_1}) \quad (1)$$

$$P_{n_1}(k,t+1) = q_1(P_0 + r_n P_{n_1} + r_b P_{b_1}) + q_0(r_n P_{n_2} + r_b P_{b_2}) \quad (2)$$

$$P_{b_1}(k,t+1) = q_0(\overline{r_n} P_{n_1} + \overline{r_b} P_{b_1}) \quad (3)$$

$$P_{n_2}(k,t+1) = q_2(P_0 + r_n P_{n_1} + r_b P_{b_1}) + q_1(r_n P_{n_2} + r_b P_{b_2}) + q_0(r_n P_{n_3} + r_b P_{b_3}) \quad (4)$$

$$P_{b_2}(k,t+1) = q_1(\overline{r_n} P_{n_1} + \overline{r_b} P_{b_1}) + q_0(\overline{r_n} P_{n_2} + \overline{r_b} P_{b_2}) \quad (5)$$

$$P_{n_i}(k,t+1) = q_2(r_n P_{n_{i-1}} + r_b P_{b_{i-1}}) + q_1(r_n P_{n_i} + r_b P_{b_i}) + q_0(r_n P_{n_{i+1}} + r_b P_{b_{i+1}}),\ 3 \le i \le m-2 \quad (6)$$

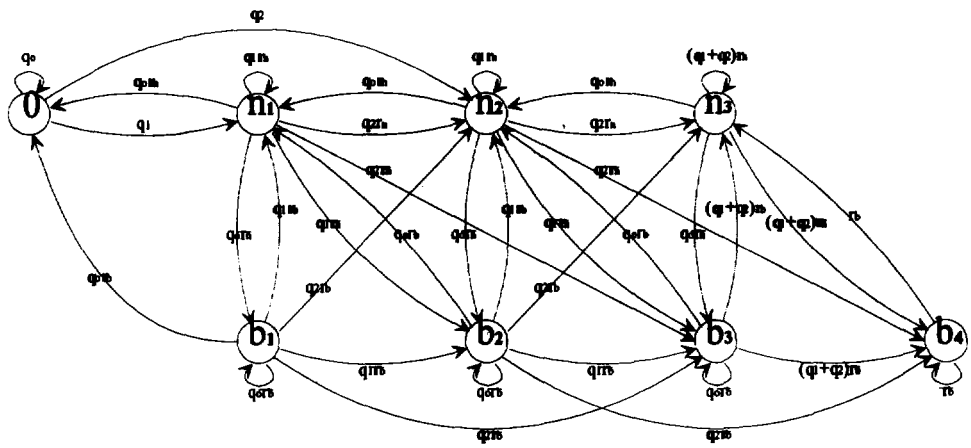
$$P_{b_i}(k,t+1) = q_2(\overline{r_n} P_{n_{i-2}} + \overline{r_b} P_{b_{i-2}}) + q_1(\overline{r_n} P_{n_{i-1}} + \overline{r_b} P_{b_{i-1}}) + q_0(\overline{r_n} P_{n_i} + \overline{r_b} P_{b_i}),\ 3 \le i \le m-1 \quad (7)$$

$$P_{n_{m-1}}(k,t+1) = q_2(r_n P_{n_{m-2}} + r_b P_{b_{m-2}}) + (q_1 + q_2)(r_n P_{n_{m-1}} + r_b P_{b_{m-1}}) + r_b P_{b_m} \quad (8)$$

$$P_{b_m}(k,t+1) = q_2(\overline{r_n} P_{n_{m-2}} + \overline{r_b} P_{b_{m-2}}) + (q_1 + q_2)(\overline{r_n} P_{n_{m-1}} + \overline{r_b} P_{b_{m-1}}) + \overline{r_b} P_{b_m} \quad (9)$$

3.2 Performance Measures

Two measures, normalized throughput and time delay, are usually used to evaluate the performance of MINs. Normalized throughput is the number of packets leaving the last stage in a network cycle, while time delay is the total time spent by a packet in the network. To show how the measures are computed for a given MIN with the MPM scheme, we first present the computation procedure.



(Fig. 3) The state transition diagram of the proposed model of 4 buffers

3.2.1 Procedure of Computation

1. Initialization at $t=0$, where q is given as an input. For the first stage, $q_1(1,0)=q(1-q)+\frac{1}{2}q^2$, $q_2(1,0)=\frac{1}{4}q^2$, $q_0(1,0)=1-q_1(1,0)-q_2(1,0)$. For all stages, $P_0(k,0)=1$, $p_{n_i}(k,0)=0(1\le i\le m-1)$, $P_{b_i}(k,0)=0(1\le j\le m)$, where $1\le k\le n$. All other variables are also set to 0.
2. $t=t+1$
3. $P_{n_i}(k,t)$, $P_{b_i}(k,t)$, and $P_0(k,t)(1\le k\le n)$ are calculated using the state equations.
4. $r_n(k,t)$ and $r_b(k,t)(\le k\le n-1)$ are calculated. Obviously, $r_n(n,t)=1$ and $r_b(n,t)=0$.
5. $T_0(k,t)$, $T_1(k,t)$, $T_2(k,t)$, $q_0(k,t)$, $q_1(k,t)$ and $q_2(k,t)(1\le k\le n)$ are calculated.
6. Throughput T and Delay D are calculated.
7. Repeat Steps 2 through 6 until T and D fully converge to stable values.

Note that the closed form of the equations of T and D are extremely difficult to obtain due to the complexity of the network and the packet movement mechanism. Therefore, as is usually done, the measures are computed by repeatedly calculating the variables until the system reaches the stable condition. We next present how the probability measures of Steps 4, 5, and 6 of the procedure above are calculated.

3.2.2 Calculation of $r_n(k,t)$

$r_n(k,t)$ is the probability that a normal packet in a buffer module of SE(k) can reach the destined buffer module in Stage$-(k+1)$ during $[t,t+1)$. For a normal packet to be able to move, thus, it must win the contention if necessary and an empty buffer space in the receiving buffer module must be available. The possible state of the head packet of the contending buffer module is 1) 0, 2) $n_i(1\le i\le m-1)$ with a different destination, 3) $n_i(1\le i\le m-1)$ with the same destination, 4) $b_i(1\le i\le m)$ with a different destination, or 5) $b_i(1\le i\le m)$ with the same destination. Recall that when two packets compete for the same destination, each packet is assumed to have the same probability to win the contention.

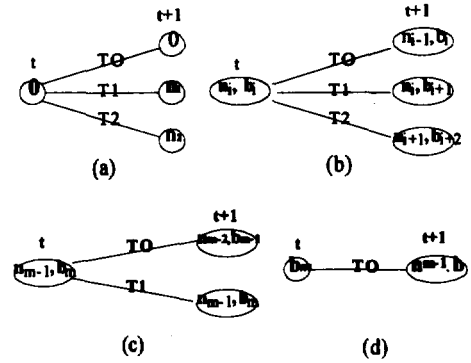
The buffer availability of the receiving buffer module in Stage$-(k+1)$ is determined by the condition of the packet movement to the module during $[t-1,t)$ in addition to the current states of the contending pair. This is the focal point of our modeling approach which allows an efficient but accurate model for MINs. Let us first consider Case 1).

1) The contending buffer module is empty.

In this case, the probability that the normal packet can move out is

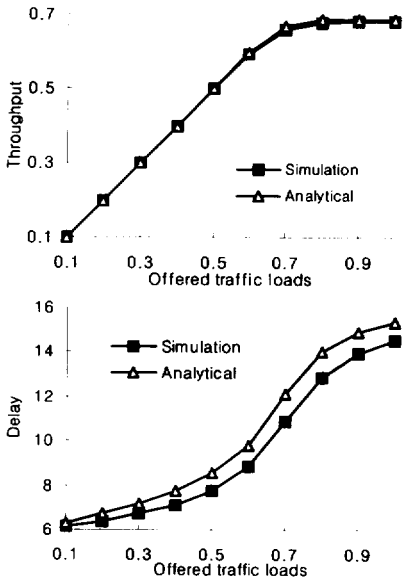
$$P_1=P_0(k,t)(T_0(k+1,t-1)\times P_A+T_1(k+1,t-1)\times P_B+T_2(k+1,t-1)\times P_C) \quad (10)$$

Here $T_0(k+1,t-1)$, $T_1(k+1,t-1)$ and $T_2(k+1,t-1)$ represent the probability that no packet, 1 packet, and 2 packets were received in the previous clock cycle, respectively. Then P_A represents the probability that a buffer space is available provided that the contending pair in the preceding stage are currently in $State-n_i$, and $State-0$ respectively, while the buffer module received no packet in the previous clock cycle. For obtaining P_A, we need to know what state the buffer module can be in now with this condition.



(Fig. 4) Source graphs obtained from the state transition diagram

First of all, from the condition that no packet was received, the possible current state is anyone of the original $2m$ states. This is clear if we check the state transition diagram of figure 3. For example, every node in the diagram has an input edge with q_0 factor(no incoming packet). Therefore any state is possible if no packet was received. For systematically deciding the possible states based on the condition of the throughput(0, 1, or 2 packets received), we develop a graph model. Observe from the diagram that, the subsequent state from *State*-0 is *State*-0, n_1, or n_2 if the throughput was T_0, T_1, or T_2, respectively. This is illustrated in figure 4(a). The possible state transitions from all other states are also shown in figure 4(b)$-$(d). We call the graph of the transitions from a state to other states *source tree*. Here we define the state at the root of each tree as *source state*, while the other states as *sink states*. Then each transition is a triple of (SO, SI, α) where SO: source state, SI: sink state, and α: throughput. For example, $(0, 0, T_0)$ represents the topmost link of figure 5(a).



(Fig. 5) Throughput and delay of 6-stage MIN with 4 buffers

The set of possible states for given T_k, S_k, is then the sink states of the transitions where $\alpha = T_k$, $k = 0, 1, 2$. From figure 5, it is easily obtained that

- $S_{T_0} = \Omega = \{0, n_i, b_j, \textit{where } 1 \le i \le m-1, 1 \le j \le m\}$
- $S_{T_1} = \Omega - \{0, b_1\}$
- $S_{T_2} = \Omega - \{0, n_1, b_1, b_2\}$

Recall that there exists another condition for P_A, which is that the contending pair are in *State*$-n_i$ and *State*-0. The set of possible states based on only this condition, S_{n0}, is Ω. This is because the state of the buffer module is independent of the states of the contending pair. For example, assume that the head packet of the buffer module in *State*$-n_i$ was not sent to it while the buffer module in *State*-0 was still in the same *State*-0 in the previous network cycle. Then the receiving buffer module can be in any state. Table 1 lists the combinations of the states of the contending pair and the corresponding set of possible states of the receiving buffer module assuming that one of the contending pair is in *State*$-n_i$.

〈Table 1〉 Possible states of a buffer module for each condition of a contending pair with a normal head packet in one of them

State of a contending pair	Possible States
1) n, 0	Ω
2) n, n (different dest.)	Ω
3) n, n (same dest.)	Ω
4) n, b (different dest.)	Ω
5) n, b (same dest.)	$\{n_{m-1}, b_m\}$

Finally, the possible states based on the conditions of contending pair and throughput are obtained as $S_{n0} \cap S_{T_0} = \Omega$. P_A is then $\frac{P(\Omega - \{b_m\})}{P(\Omega)} = 1 - P_{b_m}$, since b_m is the only state that no buffer space is available. For T_1, the possible states are $S_{n0} \cap S_{T_1} = \Omega \cap (\Omega - \{0, b_1\}) = \Omega - \{0, b_1\}$. Thus,

$P_B = \frac{P(\Omega-\{0,b_1\}-\{b_m\})}{P(\Omega-\{0,b_1\})} = \frac{\sum_{i=1}^{m-1} P_{n_i} + \sum_{i=2}^{m-1} P_{b_i}}{\sum_{i=1}^{m-1} P_{n_i} + \sum_{i=2}^{m} P_{b_i}}$. We can similarly obtain $P_C = \frac{\sum_{i=2}^{m-1} P_{n_i} + \sum_{i=3}^{m-1} P_{b_i}}{\sum_{i=2}^{m-1} P_{n_i} + \sum_{i=3}^{m} P_{b_i}}$.

2) The contending buffer module is normal with the different destination.

In this case the normal packet can proceed without contention, but the probability that the normal packet is directed to different buffer module is $\frac{1}{2}$. The probability of the packet movement is thus obtained as follows.

$$P_2 = \frac{1}{2} SP_n(k,t)(T_0(k+1,t-1)\times P_A + T_1(k+1,t-1)\times P_B + T_2(k+1,t-1)\times P_C) \quad (11)$$

Notice that the empty buffer availabilities corresponding to T_0, T_1, and T_2 are obtained by the same way as in Case 1), and it turns out that they are equal to those of P_1.

3) The contending buffer module is normal with the same destination.

Probability that the normal packets are directed to a same module is $\frac{1}{2} SP_n$. If the desired module in Stage$-(k+1)$ has enough empty buffer space to accommodate two packets, no contention occurs. Otherwise, a contention occurs, while the probability of winning the contention is $\frac{1}{2}$. The probability of this case is then

$$P_3 = \frac{1}{2} SP_n(k,t)(T_0(k+1,t-1)\times P_D + T_1(k+1,t-1)\times P_E + T_2(k+1,t-1)\times P_F) \quad (12)$$

where, $P_D = P_0 + \sum_{i=1}^{m-2} P_{n_i} + \sum_{i=1}^{m-2} P_{b_i} + \frac{1}{2} P_{n_{m-1}} + \frac{1}{2} P_{b_{m-1}}$,

$P_E = \frac{\sum_{i=1}^{m-2} P_{n_i} + \frac{1}{2} P_{n_{m-1}} + \sum_{i=2}^{m-2} P_{b_i} + \frac{1}{2} P_{b_{m-1}}}{\sum_{i=1}^{m-1} P_{n_i} + \sum_{i=2}^{m} P_{b_i}}$,

$P_F = \frac{\sum_{i=2}^{m-2} P_{n_i} + \frac{1}{2} P_{n_{m-1}} + \sum_{i=3}^{m-2} P_{b_i} + \frac{1}{2} P_{b_{m-1}}}{\sum_{i=2}^{m-1} P_{n_i} + \sum_{i=3}^{m} P_{b_i}}$

4) The contending buffer module is blocked with different destination.

Similarly, the probability that the blocked packet aims at the different module is $\frac{1}{2} SP_b$. Notice here that T_2 case is impossible, since the contending buffer module is in the blocked state. The following is the probability of this case.

$$P_4 = \frac{1}{2} SP_b(k,t)(T_0(k+1,t-1)\times P_A + T_1(k+1,t-1)\times P_B) \quad (13)$$

5) The contending buffer module is blocked with the same destination.

The probability of this case to occur while winning the contention is $\frac{1}{4} SP_b$. By referring to Table 1, the receiving buffer can be in either $State-n_{m-1}$ or $State-b_m$. Note that the receiving buffer module must have been in either $State-n_{m-1}$, $State-b_{m-1}$ or $State-b_m$ in the previous network cycle since there exists a blocked packet for it. If it was in $State-n_{m-1}$ or $State-b_{m-1}$, it must have received a packet, and thus it can be currently in either $State-n_{m-1}$ or $State-b_m$ according to what happened to the head packet, moved or blocked. On the other hand, if it was in $State-b_m$, the possible current states are again $State-n_{m-1}$ or $State-b_m$ according to the condition of head packet. As mentioned above, T_2 is not possible since the contending buffer module is in the blocked state. The probability is

$$P_5 = \frac{1}{4} SP_b(k,t)(T_0(k+1,t-1)\times P_G + T_1(k+1,t-1)\times P_G) \quad (14)$$

where $P_G = \frac{P_{n_{m-1}}}{P_{n_{m-1}} + P_{b_m}}$.

Finally, we obtain $r_n(k,t)$ as follows.

$$r_n(k,t) = \sum_{i=1}^{5} P_i \quad (15)$$

3.2.3 Calculaton of $r_b(k, t)$

$r_b(k, t)$ is the probability that a blocked packet in a buffer module of SE(k) can get to the destined buffer module in Stage $-(k+1)$ during $[t, t+1)$. It can be obtained basically using the same approach as for $r_n(k, t)$. Notice here, however, that the possible states of the receiving buffer module are always $\{n_{m-1}, b_m\}$ irrespective of the state of the contending buffer module. This is because the condition that the module under consideration is in the blocked state allows only the two states. Formally, $S_{b0} = S_{bn} = S_{bb} = \{n_{m-1}, b_m\}$. We next enumerate each of the five cases of the state of the contending buffer module.

1) The contending buffer module is empty. The probability that the blocked packet can move is

$$P_6 = P_0(k, t)(T_0(k+1, t-1) \times P_G + T_1(k+1, t-1) \times P_G) \quad (16)$$

The possible states of the receiving buffer module with this condition, while the throughput was T_0, are $S_{b0} \cap S_{T_0} = \{n_{m-1}, b_m\} \cap \Omega = \{n_{m-1}, b_m\}$. Hence the buffer availability is $\frac{P_{n_{m-1}}}{P_{n_{m-1}} + P_{b_m}}$ which is P_G. T_1 case can be similarly decided.

2) The contending buffer module is normal with the different destination. The probability that the normal packet aims at the different module is $\frac{1}{2} SP_n$. The buffer availability is obtained as in Case 1). The probability of the packet movement is then as follows.

$$P_7 = \frac{1}{2} SP_n(k, t)(T_0(k+1, t-1) \times P_G + T_1(k+1, t-1) \times P_G) \quad (17)$$

3) The contending buffer module is normal with the same destination. If the desired module in Stage $-(k+1)$ has only one free buffer, then a contention occurs. Probability of this case to occur and the blocked packet to win the contention is $\frac{1}{4} SP_n$. Then

$$P_8 = \frac{1}{4} SP_n(k, t)(T_0(k+1, t-1) \times P_G + T_1(k+1, t-1) \times P_G) \quad (18)$$

4) The contending buffer module is blocked with the different destination. The probability that the blocked packet aims at the different module is $\frac{1}{2} SP_b$. In this case, no packet could be received during $[t-1, t)$ due to the blockings of both contending modules in the preceding stage. Thus,

$$P_9 = \frac{1}{2} SP_b(k, t)(T_0(k+1, t-1) \times P_G) \quad (19)$$

5) The contending buffer module is blocked with the same destination. The probability of this case to occur and win the contention is $\frac{1}{4} SP_b$. Clearly, the probability of this case is

$$P_{10} = \frac{1}{4} SP_b(k, t)(T_0(k+1, t-1) \times P_G) \quad (20)$$

Finally, we obtain $r_b(k, t)$ as follows.

$$r_b(k, t) = \sum_{i=6}^{10} P_i \quad (21)$$

3.2.4 Calculation of T_i and q_i $(i = 0, 1, 2)$

Assuming the constant input load q, T_i's and q_i's of the first stage are

$$q_1(1, t) = q(1-q) + \frac{1}{2} q^2, \quad q_2(1, t) = \frac{1}{4} q^2,$$
$$q_0(1, t) = 1 - q_1(1, t) - q_2(1, t) \quad (22)$$

$$T_1(1, t) = q_1(1, t)(1 - P_{b_m}(1, t)) + q_2(1, t)(P_{n_{m-1}}(1, t) + P_{b_{m-1}}(1, t)) \quad (23)$$

$$T_2(1, t) = q_2(1, t)\left\{P_0(1, t) + \sum_{i=1}^{m-2}(P_{n_i}(1, t) + P_{b_i}(1, t))\right\} \quad (24)$$

$$T_0(1, t) = 1 - T_1(1, t) - T_2(1, t) \quad (25)$$

Equations for throughputs were obtained by the fact that a packet is received only when it is ready to come from a buffer module in the preceding stage and a buffer space is available for accommodating it. For other stages,

$$X(k-1,t)=SP_n(k-1,t)\,r_n(k-1,t)+SP_b(k-1,t)\,r_b(k-1,t) \quad (26)$$

$$T_2(k,t)=\frac{1}{4}\{X(k-1,t)\}^2 \quad (27)$$

$$q_2(k,t)=\frac{T_2(k,t)}{P_0(k,t)+\sum_{i=1}^{n-2}(P_{n_i}(k,t)+P_{b_i}(k,t))} \quad (28)$$

$$T_1(k,t)=X(k-1,t)\{1-X(k-1,t)\}+\frac{1}{2}\{X(k-1,t)\}^2 \quad (29)$$

$$q_1(k,t)=\frac{T_1(k,t)-q_2(k,t)\,(P_{n_{n-1}}(k,t)+P_{b_{n-1}}(k,t))}{1-P_{b_n}(k,t)} \quad (30)$$

$$T_0(k,t)=1-T_1(k,t)-T_2(k,t),\quad q_0(k,t)=1-q_1(k,t)-q_2(k,t) \quad (31)$$

Note that $X(k-1,t)$ is the probability that a packet is ready to come from a buffer module at *Stage*$-(k-1)$. The first term of Eq.(29) for $T_1(k,t)$ is the probability that only one of the contending pair has a packet to send, while the other term represents the condition that the destined buffer modules are different even though both modules have a packet. Throughputs are calculated first using the condition of the previous stage buffer modules, and then q_i's are calculated using the relation of Eqs (23) and (24). Conditions for the first and last stage are different from other stages inside the MIN. Their conditions are as follows.

1. *Stage*-1 : Since PEs do not have any buffers inside, generating a packet is independent of the network condition. Hence, probability q is specified as the initially offered traffic load to the network inputs.
2. *Stage*$-n$: Since the memory module can always accept one packet pernetwork cycle from SEs in the last stage, none of the modules in the last stage is in the blocked state. Thus, $r_n(n,t)=1$ and $r_b(n,t)=0$.

3.2.5 Throughput and Delay

Normalized throughput of a MIN is defined as the number of packets moving forward to the memory module from a buffer module in the last stage when the network reaches steady state. Hence, the normalized throughput is obtained as follows.

$$T=SP_n(n,t)\,r_n(n,t)+SP_b(n,t)\,r_b(n,t) \quad (32)$$

where t is the time for reaching the stable state.

Time delay is defined as the total time spent by a packet in the network. Using Little's theorem, the delay of the k-th stage is

$$D(k)=\lim_{t\to\infty}\frac{\sum_{i=1}^{n-1}iP_{n_i}(k,t)+\sum_{i=1}^{m}iP_{b_i}(k,t)}{T_1(k,t)+T_2(k,t)} \quad (33)$$

Time delay is then

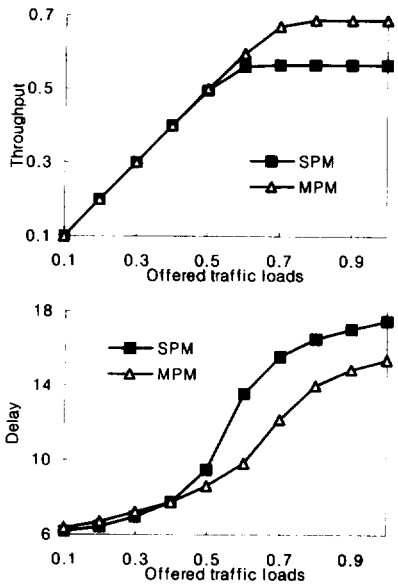
$$D=\sum_{k=1}^{n}D(k) \quad (34)$$

4. Performance Evaluation

The proposed model for MINs with MPM scheme is validated by computer simulation. Figure 5 plots the throughput and delay data obtained from the analytical model and simulation for 4-buffered MINs of 6 stages(64×64) when the offered traffic load varies from 0.1 to 1. Observe that the analytical model allows quite accurate data for entire range of offered traffic loads. The results for the bigger MINs such as 10 stages (1024×1024) show the similar trend for every input loads, and they are omitted due to space limitation.

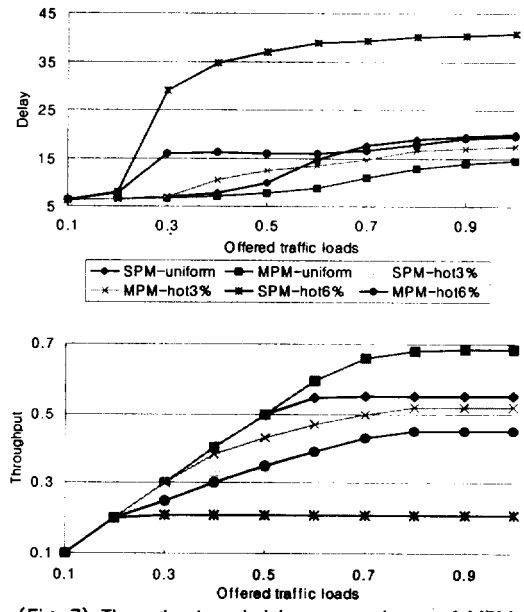
Using the model, next, the performance of the MPM scheme is compared with the SPM scheme. Figure 6 is the results of the comparison. Analytical model developed for the SPM scheme[14] was programmed for the comparison. The left in figure 6 compares the throughput of the two schemes, for a 6 stages 4-buffered MIN (64×64). The plots reveal that the MPM scheme improves the throughput up to about 30% as the offered traffic load grows to 1. Basically the same results are observed for delays as we can see from the right one in figure 6. Observe

that delay is also significantly reduced by the MPM scheme. We can also see the similar trend for the bigger size network of 10 stages. They are omitted here. Notice that the MPM scheme consistently outperforms the SPM scheme.



(Fig. 6) Throughput and delay comparison of MPM and SPM schemes for 6-stage MIN with 4 buffers

There are little differences when the traffic load is lower than 0.5. This is because, with this low traffic, not many packets are blocked. Therefore the effect of multiple packet movement is not so significant. However, if the traffic is not uniform, the MPM scheme will be again very effective to allow a reasonable network performance by reducing the congestion. This is studied in figure 7. In figure 7, throughputs of the two schemes are compared by simulation when there exist some nonuniform traffic due to hot spot. Here the processors make a fraction h of their requests to a hot memory module, while the remaining $(1-h)$ of their requests are distributed uniformly over all memory modules including the hot one. Observe from the figure that, when $q=0.9$ and h is 3%, the throughput of MPM scheme is about 55% higher than that of SPM scheme. For the same input load, the difference was about 23% if there was no nonuniform traffic. We see more improvement of 115% when the nonuniform traffic increases as $h=6\%$. In other words, for the hot rate of $h=0$, 3, and 6% the throughputs of SPM scheme are 0.55, 0.34, and 0.21. The corresponding values for MPM scheme are 0.68, 0.52, and 0.45. As expected, MPM scheme displays consistently high performance irrespective of the existence of nonuniform traffic. Basically the same results are obtained in the delay comparisons as shown in the right one in figure 7.



(Fig. 7) Throughput and delay comparisons of MPM and SPM schemes with hotspot for 6-stage MIN with 4 buffers

5. Conclusion

We have identified the importance of the number of packet movements in one network cycle for multiple-buffered MINs and the position of the buffers in each switching element. In earlier designs, typically only one packet movement was assumed. To efficiently and correctly evaluate the proposed multiple packet movement(MPM) scheme with

output buffering, a new analytical model has also been developed which can systematically model the network operation while realistically taking the blocked packets into account. Computer simulation verified that the model is accurate for practical size and operational conditions of MINs. When compared with the traditional single packet movement(SPM) scheme, the MPM scheme always outperforms it, and it is more significant when the traffic is relatively high or nonuniform. We are investigating the effectiveness of the proposed scheme under various operational conditions of MINs such as ATM switching.

References

[1] C.L. Wu and T.Y. Feng, "On a class of Multistage Interconnection Networks," IEEE Trans. Computers, Vol.C-29., pp.694-702, August 1980.

[2] CCITT Recommendation I.121, "Broadband aspects of ISDN," Blue Book, Vol.III.7, Geneva, Switzerland, 1989.

[3] J.S. Turner, "Design of an integrated services packet network," Ninth Data Commun. Symp., in ACM SigComm Comput. Commun. Rev., Vol.15, pp.124-133, Sept., 1985.

[4] A. Varma and C.S. Raghavendra, "Performance analysis of a redundant-path interconnection network," in Proc. 1985 Int. Conf. Parallel Processing, pp.474-479, 1985.

[5] H. Rudin, "The ATM-Asynchronous Transfer Mode," Computer Networks and ISDN Systems, Vol.24, pp.277-278, 1992.

[6] J.H. Patel, "Performance of processor-memory interconnections for multiprocessors," IEEE Trans. on Computers, Vol.c-30, pp.771-780, Oct., 1981.

[7] C.P. Kruskal and M. Snir, "The performance of multistage interconnection networks for multiprocessors," IEEE Trans. on Computers, Vol.c-32, pp.1091-1098, Dec., 1983.

[8] M. Kumar and J.R. Jump, "Performance of unbuffered shuffle-exchange networks," IEEE Trans. on Computers, Vol.c-35, pp.573-578, June 1986.

[9] D.M. Dias and J.R. Jump, "Analysis and simulation of buffered delta networks," IEEE Trans. Computers, Vol.c-30, pp.273-282, Apr., 1981.

[10] Y.C. Jenq, "Performance analysis of a packet switch based on single-buffered banyan network," IEEE J. Select. Areas Commun., Vol. SAC-3, pp.1014-1021, Dec., 1983.

[11] H.S. Yoon, K.Y. Lee and M.T. Liu, "Performance analysis of multibuffered packet-switching networks in multiprocessor systems," IEEE Trans. on Computers, Vol.c-39, pp.319-327, March 1990.

[12] T.H. Theimer, E.P. Rathgeb and M.N. Huber, "Performance analysis of buffered banyan networks," IEEE Trans. on Commun. Vol.c-39, pp. 269-277, Feb., 1991.

[13] S.H. Hsiao and C.Y.R. Chen, "Performance analysis of single-buffered multistage interconnection networks," in Proc. Third IEEE Symp. on Parallel and Distributed Processing, pp.864-867, Dec., 1991.

[14] Y. Mun and H.Y. Youn, "Performance Analysis of Finite Buffered Multistage Interconnection Networks," IEEE Trans. on Computers, pp.153-162, Feb., 1994.

[15] J. Ding and L.N. Bhuyan, "Performance evaluation of multistage interconnection networks with finite buffers," in Proc. 1991 Int'l. Conf. on Parallel Processing, pp.592-595, 1991.

[16] H.Y. Youn and Y. Mun, "On Multistage Interconnection Networks with Small Clock Cycles," IEEE Trans. on Parallel and Distributed Systems, pp.86-93, Jan., 1995.

[17] H.Y. Youn and C. Chevli, "A Local Blocking Scheme for Performance Enhancement of MINs," in 1991 Int'l Symp. on Applied Computing, pp. 273-282, April, 1991.

[18] G.F. Pfisher and A.V. Norton, "Hot Spot' Contention and Combining in Multistage Interconnection Networks," IEEE Trans. on Computers, Vol.C-34., No.10, pp.943-948, Oct., 1985.

[19] N.F. Tzeng, "Designing of a Novel Combining Structure for Shared-Memory Multiprocessors," in Proc. IEEE 1989 Int'l Conf. on Parallel Processing, pp.I1-I8, 1989.

[20] S.L. Scott and G.S. Sohi, "Using feedback to control tree saturation in multistage interconnection networks," in Proc. the 16th Annual Int'l. Symp. on Computer.

추 현 승

e-mail : choo@yurim.skku.ac.kr

1988년 성균관대학교 이과대학 수학과 졸업(학사)

1990년 텍사스 주립대 달라스 소재 대학원 전자계산학과 (공학석사)

1996년 텍사스 주립대 알링턴 소재 대학원 전산공학과 (공학박사)

1997년~1998년 특허청 심사4국 컴퓨터심사 담당관실 심사관

1998년~현재 성균관대학교 전기전자 및 컴퓨터공학부 전임강사

관심분야 : ATM, 병렬 및 분산 처리, 알고리즘 해석, 고속통신망 등

박 경 린

e-mail : glpark@cheju.cheju.ac.kr

1986년 중앙 대학교 전자계산학과 졸업(학사)

1988년 중앙 대학교 전자계산학과 대학원(공학석사)

1992년 텍사스 주립대 알링턴 소재 전산공학과 대학원(공학석사)

1997년 텍사스 주립대 알링턴 소재 전산공학과 대학원 (공학박사)

1998년~현재 제주대학교 전산통계학과 전임강사

관심분야 : 분산/병렬 처리 시스템, 오류 허용 시스템, 성능 평가 등