論文98-35S-10-7

입력 트래픽의 특성에 따라 복사 수가 제어되는 ATM 멀티캐스트 스위치 복사 망

(The copy networks controlling the copy number according to the fluctuations of the input traffics for an ATM Multicast Switch)

白 政 勳 *, 林 濟 鐸 **

(Jung Hoon Paik and Chae Tak Lim)

요 약

본 논문에서는 기존의 멀티캐스트 패킷 교환기의 문제점에 대한 개선 방안을 제안한다. 오버플로우에 의해 야기 되는 입력포트의 공평성 문제는 임의의 입력 포트로부터 복사 수를 가산 할 수 있는 동적 시작점 결정기에 의해 해결된다. 시작점은 매 타임 슬롯 마다 입력 버퍼의 점유도와 이전 타임 슬롯의 오버플로우를 기반으로 가변 된다. 입력 버퍼의 점유도를 이용함으로서 제안된 복사망은 기존의 방식에 비해 입력 트래픽의 변동에 대하여 우수한 적응성을 제공한다. 동적 시작점 결정기는 입력 트래픽의 양에 따라 복사 요청의 수를 제어하며 이것은 복사망의 전체 스루풋을 제고하는 필수적인 기능 이다. 오버플로우 발생시에 멀티캐스트 스위치의 스루풋을 향상시키는 호-분할(call-splitting) 방식도 동적 시작점 결정기에 의해 제공된다. 동적 시작점 결정기의 하드웨어는 고속 동작에 적합한 단순한 구조로 도출된다. 제안된 방식의 성능 평가를 위해 다양한 트래픽에 대한 모의 실험 결과가 제공된다.

Abstract

In this paper, several improvements to a copy network proposed previously for multicast packet switching are described. The improvements provide a solution to some problems inherent in multicasting. The input fairness problem caused by overflow is solved by a dynamic starting point decider(DSD), which can calculate running sums of copy requests starting from any input port. The starting point is changed adaptively in every time slot based on both the fill level of the input buffers in current time slot and the overflow situations of the previous time slot. Using the fill level of the input buffers, the copy network shows better adaptability to the fluctuation of input traffics than the conventional network. The DSD also provides the function of regulating overall copy requests according to the amount of input traffics. This is an essential function in improving overall throughputs of the copy networks. The throughput of a multicast switch can be improved substantially if partial service of copy request is implemented when overflow occurs. Call-splitting can also be implemented by the DSD in a straightforward manner. The hardware for the DSD is derived with the objective of simple architectures for the high speed operation. Simulation study of the copy network under various traffic conditions is presented to evaluate its performance.

* 正會員, 斗源工科大學 電波通信科
(Dept. of Radio Communication, Doowon Technical College)

** 正會員, 漢陽大學校 電子工學科
(Dept. of Elec. Eng., Hanyang University)

接受日字: 1998年7月28日, 수정완료일: 1998年10月2日

I. Introductions

With the advent of the *broadband integrated services digital network*(B-ISDN), the types of the both connections and traffics that the networks should support are various. That is, B-ISDN should support applications of different connectivity requirements such as point-to-point, point-to-multipoint, and multipoint-to-multipoint connections. In addition, it should support services of vastly different data rates. To handle this kind of connections, multicasting functions are required in switches. The architectures that have been proposed for multicast ATM switches can be classified into two categories based on their cell-replication methods : 1) multicast tree type[1][2][3], and 2) broadcast type[4][5][6][7]. In the former case, multicasting is accomplished by a copy network followed by a point-to-point switch. The copy network generates the copies requested by incoming packets, and the point-to-point switch routes the replicated copies to their final destinations.

This paper describes several improvements to the non-blocking copy network proposed in [1] and [8], in which packet replications are accomplished by an encoding process and a decoding process. As shown in Fig. 1, it consists of a running adder network(RAN), a set of *dummy address encoders*(DAE's), a concentrator, a *broadcast banyan network*(BBN), and a group of *trunk number translators*(TNT's). The encoding process is carried out by the *RAN* and a set of *DAE*'s at the outputs of the *RAN*. The sequence of running sums of the copy numbers requested by incoming multicast packets is first calculated in the *RAN* and then pairs of adjacent running suns are used by the *DAE*'s to form a pseudo-header for each multicast packet. The header contains a dummy address interval specified by the *minimum*(MIN) and the *maximum*(MAX) of the output port address interval and an *index reference*(IR). *IR* is used to distinguish the packet copies having the same multicast call. In the decoding process, the *BBN* will perform the packet replications by *the Boolean interval splitting* algorithm[1] based on the dummy address interval(MIN, MAX), and the *TNT*'s will determine the destination address of each replicated packet based on its copy index and broadcast channel number.

In copy networks design, an inherent problem is the overflow of the copy requests in the *RAN*, which occurs when the total number of copy requests exceed the capacity of the copy network. In Fig. 1, the copy requests from port 0 to port 3 are 2, 0, 3 and 4 respectively and the size of the copy network is 8. So, the copy request from port 3 results in the overflow. The overflow problem may degrade the throughput of the copy network, and introduce unfairness among incoming multicast packets. The utilization of the copy network would be improved with *call-splitting* in which the copy requests that cannot be copied within one time slot are partially serviced[2][3][8]. As illustrated in Fig. 1, if input port 3 receives a partial service with *call-splitting*, a total of eight copies rather than five copies could be generated in one time slot.

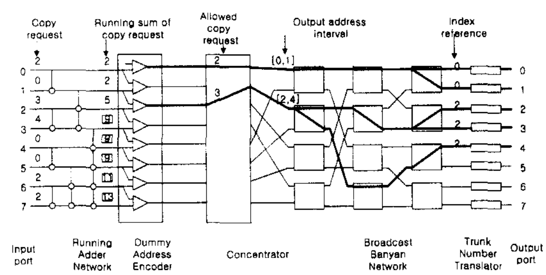


그림 1. 8x8 블록킹 없는 복사 망
Fig. 1. An 8x8 nonblocking copy network.

Although the methods improve throughputs, they don't provide complete solutions under the B-ISDN environments because the fluctuation of the traffic is so high in B-ISDN. Therefore, it is preferable function in the copy network that controlling the overall copy requests according to the amount of the input traffics dynamically. That is, the copy network adapting to the input traffics dynamically is required, and such copy network is suggested in this paper. Another undesirable consequence of overflow is the input fairness problem, which arises due to the fixed structure

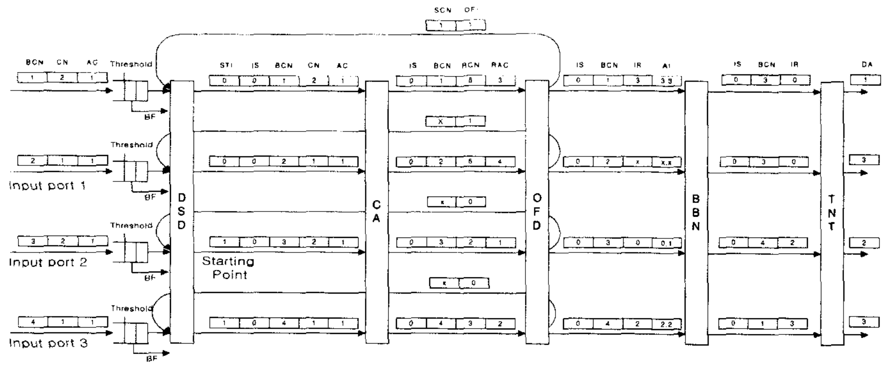


그림 2. 제안된 4x4 블록킹 없는 복사 망
Fig. 2. An 4x4 suggested nonblocking copy network.

of the *RAN*. Since the calculation of running sum always starts from input port 0 in every time slot, lower numbered input ports have higher service priorities than the higher numbered ports, especially when the overflow occurs frequently. The approach for improving the fairness is performed in the dual-adder structure[1]. But, as it is configured with two identical adders, the complexity rather increases. In addition to it, the relative unfairness among the input ports remains unsolved. And some other method[8],[9] alleviates this problem, but do not provide complete solutions. In determining the starting point, the copy network should also consider the input traffics. As an example, in case that the traffics are concentrated on some input ports, the concentrated input ports should be serviced with high priority to lower the packet loss rate. So, the starting point should be given to the concentrated input ports under the condition. Therefore, the essential factor in resolving the problem of both the overflow and the unfairness in copy network is the characteristic of the input traffics. Another requirement to the copy network in the broadband environments is high-speed operation. For the high-speed operation, it should be structured with hardware logic. In addition, it should be constructed with simple structures. The copy network suggested in this paper is structured and operated with this consideration and features both the high-speed operation and the high adaptability to diverse traffics.

The remaining paper is organized as follows. Section II describes the structure and the operation of proposed copy network. In this section, the hardware logic for the copy network targeting the high-speed operation is also derived. Section III describes a queueing model of the copy network and presents simulation results. Finally, the conclusion is provided in Section IV.

II. The Architecture of Copy Network

The structure of the suggested copy network and the associated packet format are given in Fig. 2. The copy network consists of the *input buffer*, *dynamic starting point decider*(DSD), *cyclic adder*(CA), *overflow detector*(OFD), *broadcast Banyan network*(BBN), *trunk number translator* (TNT), and the feedback path from *OFD* to *DSD*. The header of the arriving packets to the input ports consists of *broadcasting channel number* (BCN), *copy number*(CN), and *activity bit*(AC). *BCN* is an identifier for the packets belonging to the same multicast call. *CN* means the number of copy request and *AC* is set to 1 or 0 to indicate that the input packet is active or idle, respectively. Table 1 shows the routing table built in *TNT* for a multicast call. In the table, *destination address*(DA) indicates the destination ports for each copied packets and *copy index*(CI) is an identifier for each copied packets with the same

BCN.

표 1. 멀티캐스트 호에 대한 경로표
Table 1. Routing table for a multicast call.

BCN	*CI*	*DA*
1	*0*	*3*
1	*1*	*2*
2	*0*	*2*
3	*0*	*1*
3	*1*	*3*
4	*0*	*2*

The overall operation of the copy network is as follows. The input packets are arrived in input buffer where threshold is set. When the occupancy of the input buffer passes the threshold, *buffer full*(BF) signal is generated and sent to the *DSD*. The *DSD* has the function of determining the starting point for the sum of the copy requests. It is noted that the determination of the starting point affects both the throughput and the fairness in the copy network. In deciding the starting point, the proposed copy network considers both the fill level of the input buffer to reflect the characteristic of the input traffics and the overflow in the previous time slot. The fill level is indicated with the *BF* signal and the overflow of the previous time slot is indicated with the *OFI* field of the message from *OFD*. The copy network also controls the overall number of copy requests, which are becoming serviced in current time slot according to the occupancy of the input buffer. Therefore, the copy network provides the function of regulating the input traffics. The *CA* adds the copy requests from the input port that is determined as a starting point by the *DSD*. The *OFD* detects the overflow from the summing results and sends the *DSD* the number of copy requests which are not serviced in case that the overflow happens. In the *BBN*, packets are replicated and routed to the output ports.

1. Dynamic Starting Point Decider(DSD)

1) Operations of *DSD*

The *DSD* decides the starting point for the summing process. Both the *overflow indication*(OFI) field from *OFD* and the *buffer full indication*(BFI) signal from the input buffers are used in determining the starting point as illustrated in Fig. 2. *OFI* indicates whether the overflow have occurred in the previous time slot. The *BFI* is generated when the occupancy of the input buffer is passed the threshold. The *DSD* indicates the next starting point with *starting point indication*(STI) field and generates *index shift*(IS) with the default value of 0. If port k is determined as the starting point, *STI* is set to 1 for the ports which span from port k to port $N-1$ where N is the size of the copy network and 0 for other ports. That is, STI_i = *0 for 0 i k-1 and* STI_i = *1 for k i N-1. IS* is used in deriving *CI* in *TNT*. The procedure of deciding the starting point is as follows.

- Case 1 : All BFI^t are 0 and all OFI^{t-1} are 0

In this case, overflow in current time slot, *t*, as well as buffer full in previous time slot, *t-1*, doesn't happen. Therefore, all the copy requests have been fully served in previous time slot. So, the *head of line*(HOL) packets in the input buffers are removed and the next packets are move to the *HOL* of the buffers. The starting point is moved to next input port for the fairness. That is, starting point is determined with the round-robin manner. The implementation of the scheme is given to next sub-section and it features simple structures which is suitable for the high-speed environments. As all the copy number are processed, there is no need to change *IS*

- Case 2 : All BFI^t are 0 and some OFI^{t-1} are 1

This is the case where the overflow occurred in previous time slot and no buffer-full state happens. So, all copy requests are not served. As the occupancy of the input buffer is below the threshold in this case, it is preferable for the improvement of the throughput that the residual copy requests that were not served in previous time slot are serviced in current time slot. For fairness, starting point should be the first port encountered the overflow in previous time slot, and it is detected with the port i which satisfies the condition that $OFI_i^{t-1}=1$ and $OFI_{i-1}^{t-1}=0$ cyclically. As the information on the actually served number of copy requests of port i in previous time slot is returned with *served packet number*(SPN) from *OFD*, *CN* indicating the number of copy request in a time slot should be updated. Also, *STI* and *IS* should be changed to reflect the overflow situations. This change is given by $CN_k^t = CN_k^{t-1}$ $SPN_k^{t-1}(k = i)$, $CN_k^t = CN_k^{t-1}$ $(i+1 \leq k \leq N-1)$, $STI_k^t = 0(\ 0 \leq k \leq i-1)$, $STI_k^t = 1(\ i \leq k \leq N-1)$, $IS_k^t = SPN_k^{t-1}(k = i)$, $IS_k^t = IS_k^{t-1}(i+1 \leq k \leq N-1)$.

- Case 3 : some BFI^t are 1.

This is the case where some input buffers are imminent to the fullness of the occupancy. When some input buffers are severely overloaded transiently, it is an appropriate strategy in alleviating the packet loss rate to give higher service priority to the buffers. So, regardless of the overflow in previous time slot, starting point is given to the overloaded input buffer. The *HOL* packets that were not serviced in previous time slot are not further processed. That is, they are discarded, and new packets are moved to the *HOL*. So, there is no change in *IS*. When only one buffer is in the state of fullness, the starting point is given to the port. In the case that more than one input buffers are imminent to the state of fullness, one among the associated input ports is selected to be the starting port in round-robin manner.

2) Hardware Structures of *DSD*

As illustrated above, *DSD* should be operated with round-robin manner for both case 1 and case 3. For the high speed environments, the round-robin manner should be implemented only with hardware. In addition, it should be configured with simple structures that result in the decrease of the number of logic gates which is indispensable factor in decreasing heat and propagation delays between gates. Simple structures for the round robin scheme can be implemented based on a bi-directional arbiter. As both case 1 and case 3 are structured identically, only the structure for the case 3 is derived. The structure is shown in Fig. 3.

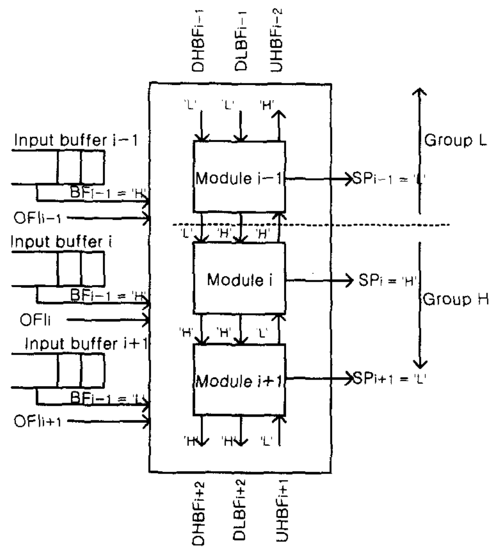


그림 3. 시작점 결정기의 구조

Fig. 3. The structure of SPD for the buffer-full state.

The modules are allocated for each input buffer and three control signals, *DHBF*(*Down High-group Buffer Full*), *DLBF*(*Down Low-group Buffer Full*), and *ULBF*(*Up High-group Buffer Full*) are used. Modules are classified into two groups, Group *H* and Group *L*. The module that has just selected as a starting point, and all modules above it are formed as Group *L* while all lower modules are formed as Group *H*. Group *H* has higher priority than Group *L*, and in each Group, the upper module has higher priority than lower. This scheme results in the round robin manner from higher input ports to lower input ports cyclically. The signals are transmitted on each bus lines. *DHBF* and *DLBF* are down-stream signals while *UHBF* is the up-stream signal. If buffer-full state happens to a module in Group *H*, it generates *DHBF* with high value as $DHBF_{i+1}$ in Fig. 3. Otherwise, it only passes the signal which is received from the upper module to lower module as $DHBF_{i+2}$ in Fig. 3. So, *DHBF* signal is to select the highest buffer-full module in Group *H*. *DLBF* functions like *DHB*F for Group *L*. *UHBF* is generated by a module of Group *H* to which buffer-full state happens as $UHBF_{i-1}$ in Fig. 3. A module in Group L identifies with this signal whether Group H has the buffer-full modules. The operation of each module depends on the group to which the modules belong. Therefore, it should be described separately. When a module (called tagged module thereafter) is in the Group H, four cases should be considered. The first case is that the buffer-full states happen to the modules that are above the tagged module. This is indicated with the high value of $DHBF$ signal. As the upper module has higher priority, the tagged module can not be starting point. In this case, the starting point is given to a module which is above the tagged module. As the tagged module is below the next starting point, the group of it is changed to high. The second case is that the tagged module is the highest buffer-full module in the Group H. This situation is indicated with the low value of $DHBF$ signal and high value of BF signal. In this case, the tagged module can be the starting point, and its next group is transited to low. The third case is that the buffer-full states are only happened to the modules of the Group H that are below the tagged module. It is indicated with the low value of $DHBF$ and the high value of $UHBF$. In this case, the tagged module cannot be the starting point and its next group is changed to low. The fourth case is that buffer-full states are only happen in *Group L*. This is indicated with the high value of $DLBF$ and the low value of both $DHBF$ and $UHBF$. In this case, tagged module cannot be the starting point and its next group is high. When the tagged module belongs to the Group L, the following cases can be classified. The first case is that Group H has the module of buffer-full state. This is indicated with the high value of $UHBF$. In this case, the tagged module cannot be starting point, and its state remains unchanged. The second case is that buffer-full states are happened among the upper modules than the tagged module under the condition that no buffer-full states are happened to the Group H. This is indicated with the low value of the $UHBF$ and the high value of $DLBF$. In this case, tagged module also cannot be starting point and its next group is changed to high. The third case is that tagged module is the highest module of buffer-full state in Group L and no buffer-full states are happened in Group H. This is the case where tagged buffers can be starting point and its next group is Group L. This is summarized in Table 2. Based on the Table 2, the logic circuit for a module is structured as Fig. 4. As it is indicated, the module is constructed with very simple structure. Therefore, the proposed copy network features high-speed operation and high adaptability to the bursty traffic. This property is an important factor for handling the broadband multimedia traffics.

표 2. 모듈에 대한 논리표

Table 2. A logic table of a module for the buffer-full state.

Input					Output				
PG	BF_i	$DHBF_i$	$DLBF_i$	$UHBF_i$	SP_i	$DHBF_{i+1}$	$DLBF_{i+1}$	$UHBF_{i-1}$	NG
H	X	H	X	X	L	$DHBF_i$	$DLBF_i$	$UHBF_i$	H
H	H	L	X	X	L	H	$DLBF_i$	H	L
H	L	L	X	H	L	$DHBF_i$	$DLBF_i$	$UHBF_i$	L
H	L	L	H	L	L	$DHBF_i$	$DLBF_i$	$UHBF_i$	H
L	X	X	X	H	L	$DHBF_{i+1}$	$DLBF_i$	$UHBF_i$	L
L	X	X	H	L	L	$DHBF_i$	$DLBF_i$	$UHBF_i$	H
L	H	X	L	L	H	$DHBF_i$	H	$UHBF_i$	L
L	L	X	L	L	L	$DHBF_i$	$DLBF_i$	$UHBF_i$	L

* PG : Present Group, NG : Next Group

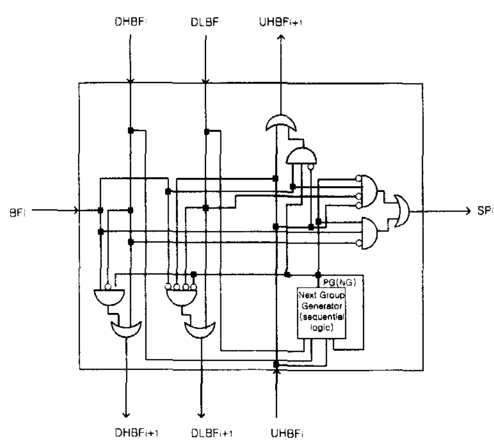


그림 4. 시작점 결정기의 논리도

Fig. 4. The structure of SPD for the buffer-full state

2. Cyclic Adder(*CA*)

CA generates *running sum of copy number*(RCN) and *running sum of activity*(RAC) which are cyclically summing result of *CN* and *AC* respectively from *STI*. As *DSD* sets *STI* field to 1 for the ports which span from the starting port to the last port, *CA* should be structured to add the *CN* from the starting point cyclically according to *STI*. The structure and associated header fields are shown in Fig. 5. In Fig. 5, as port *2* is configured as the starting point, *CA* performs the summing process from port *2* cyclically.

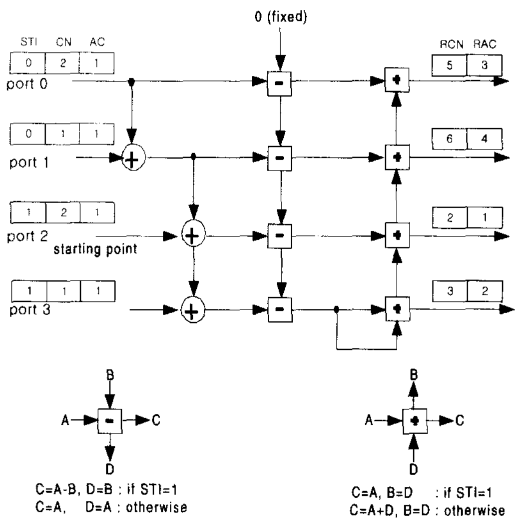


그림 5. 순환 가산기의 구조
Fig. 5. The structure of the cyclic adder.

3. Overflow Detector(*OFD*)

OFD detects overflow based on the *RCN* field from the *CA* and sends back the *SPN* to the *DSD*. The overflow happens when the *RCN* is larger than the size of the copy network. That is, when total number of copy requests in a time slot are larger than the size of the copy network, the overflow happens in the copy network. *OFD* sets the *OFI* field to 1 for the ports in which overflow happens. *SPN* represents the number of copies allowed for the packet encountered the overflow for the first time. This acknowledgment information is used for performing call-splitting in *DSD*. In Fig. 6, since the value of *RCN* from the port *3* and *0* are 3 and 5 respectively, two copy requests of port 0 results in the overflow. Call-splitting only serves one among two copy requests of port 0. So, *SPN* is set to one and sent back to the *DSD*. *OFD* also .generates *address interval*(*AI*) and *index reference*(*IR*) based on the *RCN*. *AI* is the address interval to which the copied packets are destined and it is defined as a *minimum address*(*MIN*) and a *maximum address*(*MAX*). *IR* is used in identifying packets belonging to the same *BCN*. *AI* and *IR* is determined from both the *RCN* and *RAC* as follows.

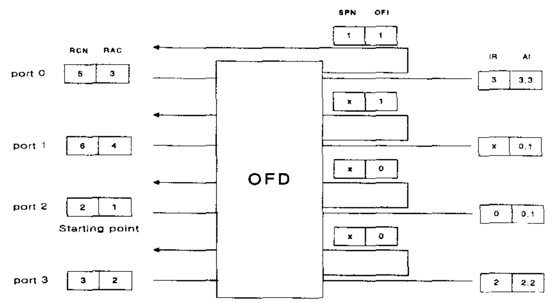


그림 6. 오버플로우 검출기의 동작
Fig. 6. The operation of the overflow detector

- Case 1 : $RCN_i \leq N$

This is the case where all the requested copy numbers are to be copied with no overflow. So, *AI* and *IR* are given by

$$MIN_i = \begin{cases} 0 & \text{If } RAC_i = 1 \\ RCN_{i-1} & \text{Otherwise} \end{cases}$$

$$MAX_i = RCN_i - 1$$

$$IR_i = MIN_i$$

- case 2 : $RCN_i > N$

In this case, port i is one among the overflowed input ports. So, *OFI* should be set to 1 to indicate an overflow. *AI*, *SPN*, and *IR* are determined as follows.

- case 2-1 : $RCN_{i-1} < N$

In this case, port i is the lowest numbered port among the overflowed ports, and only part of requested copy number of port i are served with following values.

$SPN_i = N - RCN_{i-1}$
$MIN_i = RCN_{i-1}$, $MAX_i = N-1$
$IR_i = MIN_i$

- case 2-2 : $RCN_{i-1} \geq N$

This is the case where overflow happens in the port i, but not for the first time. As no service can be given to the port i, SPN_i should be set to 0 and both AI and IR are not meaningful.

4. Broadcast Banyan Network(*BBN*)

BBN performs the packet replication in addition to the packet routing which are based on the *boolean interval splitting* algorithm. Fig. 7 shows the procedure of four replications and routing to port 3,4,5, and 6.

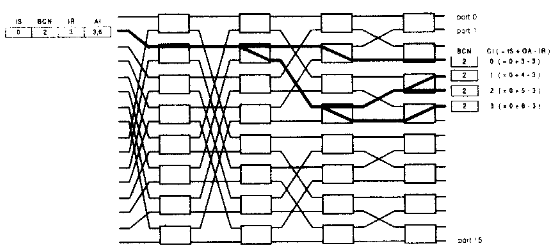


그림 7. 16x16 방송형 반얀 망
Fig. 7. A 16x16 Broadcast Banyan network.

5. Trunk Number Translator(*TNT*)

As there is a routing table like a Table1 in *TNT*, *CI* should be built from *IS* and *IR* in order to identify each copied packets belonging to the same *BCN*. This is performed with equation (1) where *output address*(*OA*) indicates the destination ports of each copied packet. The example of the translation is shown in Fig. 6.

$$CI = IS + OA - IR \tag{1}$$

Ⅲ. Modeling and simulation

1. Modeling

For the throughput and packet loss rate of the copy network, *virtual-output buffering model* [9] is used. In this model, all input packets are assumed to be duplicated to the required number of copies before they enter the buffer, and then the copied packets are stored in the buffer called virtual-output buffer. Thus, the complete network can be modeled to be a single-input stream, N-server queueing system with finite buffer size and fixed one time-slot service time. Knowing the buffer size of the shared-input buffer, which is denoted by Nb where N is the network size and b is the average buffer size per input line, the equivalent buffer size for the copied, denoted by b_{eqv}, can be estimated as $b_{eqv} = NxbxE(C)$, where $E(C)$ is the average number of copied required by an input packet. The total number of copy arrivals from all input packets per time slot, denoted by Y, can be represented as

$$Y = A_1 + A_2 + \dots\dots\dots + A_N \tag{2}$$

In equation (2), A_i is a random variable representing the number of copies generated by the ith input regardless of whether it is active or idle. The probability distribution of A_i can be represented

$$A_i(k) = P(A_i=k) = \begin{cases} 1-\rho, & k=0 \\ \rho C(k), & k=1,2,\cdots,M \end{cases} \tag{3}$$

where ρ represents the probability of arrival of an input packet in a time slot and C is the number of copies requested by an input packet. The probability distribution of Y, denoted by $Y(k)$, is the discrete convolution $A_i(k)$, which is shown in equation (4).

$$Y(k=n) = A_1(k) * A_2(k) * \dots\dots\dots * A_N(k) \quad n=0,1,\dots\dots,NM \tag{4}$$

In (4), M means the maximum allowable number of copies requested by an input packet. The input distribution function $P(W=i)$ of the equivalent *virtual output buffer* is, therefore, obtained by

$$p(w=i) = \sum_{j=i}^{NM} p(Y=j)p(W=i/Y=j)$$
$$= \sum_{j=i}^{NM} p(Y=j)\binom{j}{i}\left(\frac{1}{N}\right)^{i}\left(1-\frac{1}{N}\right)^{j-i} \tag{5}$$

where W is the number of packet of the equivalent virtual output buffer. With the random variable Q representing the number of copies in the queue at the end of a time slot, the state transition probabilities of the queue length $P_{ij}=P(Q_m=j\,|\,Q_{m-1}=j)$ are determined by

$$P_{ij}=\begin{cases} w_0+w_1 & i=0,\ j=0 \\ w_0 & 1\le i\le bE(C),\ j=i-1 \\ w_{j-i+1} & 0\le i\le j,\ 1\le j\le b-1 \\ \sum_{m=j-i+1}^{N} w_m & 0\le i\le j,\ j=bE(C) \\ 0 & otherwise \end{cases} \qquad (6)$$

where $w_i=P(W=i)$ and b is the mean buffer size per input port. The steady-state queue length in the buffer can be directly obtained from the *Markov* chain balance equations as in (7).

$$Q_0=p(Q=0)=\frac{1}{1+\sum_{n=1}^{bE(c)}\frac{Q_n}{Q_0}}$$

$$Q_1=p(Q=1)=\frac{1-w_0-w_1}{w_0}Q_0 \qquad (7)$$

$$\vdots$$

$$Q_n=p(Q=n)=\frac{1-w_1}{w_0}Q_{n-1}-\sum_{k=2}^{n}\frac{w_k}{w_0}Q_{n-k} \quad 2\le n\le bE(C)$$

With the above modeling, the throughput and packet loss performance of the copy network can be estimated. In copy network, if and only if the buffer is empty and there are no new arrivals, the output is zero; otherwise, the number of outputs is always equal to one. So, the throughput can be estimated by

$$TP = 1-Q_0w_0 \qquad (8)$$

Dividing the throughput by the effective input load $_{eff}$, the packet loss probability is obtained like (9). In (9), the

$$P_{loss}=1-\frac{TP}{\rho_{eff}} \qquad (9)$$

effective input load is defined as the average number of copy packets arrived at input port per time slot like (10)

$$\rho_{eff} = E(C) \qquad (10)$$

2. Simulation and Analysis

The simulation result on the packet loss rate as a function of an effective load for the conventional copy network [1] and the suggested copy network is shown in Fig. 8 and Fig. 9 and more detailed data is listed in Table 3. The packet loss rate is derived from dividing discarded copy numbers by requested total copy numbers in copy network.

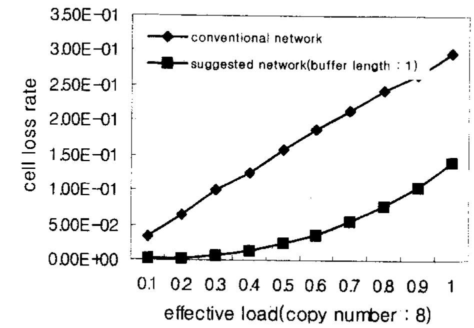


그림 8. 실효 로드에 대한 패킷 손실율
Fig. 8. Packet loss for the effective load.

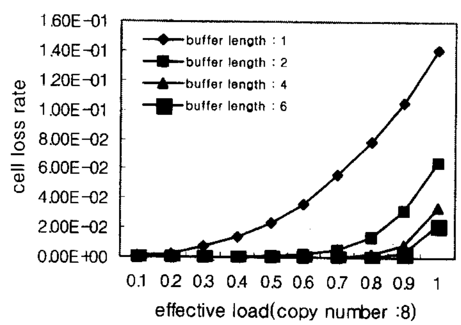


그림 9. 버퍼 길이에 대한 패킷 손실 율
Fig. 9. Packet loss ratio for the buffer length.

Table 3 shows that the suggested copy network only with 1 of buffer length improves the packet loss rate about 2 to 27 times over the conventional copy network.

표 3. 패킷 손실 율
Table 3. Packet loss rate.

Loss / Effective load	Conventional copy network	Suggested copy network			
		Buffer length 1	Buffer length 2	Buffer length 4	Buffer length 6
0.1	0.034032	0.001256	0	0	0
0.2	0.065381	0.002513	0	0	0
0.3	0.099269	0.007016	0.000066	0	0
0.4	0.124653	0.013484	0.000175	0	0
0.5	0.158867	0.023532	0.000916	0	0
0.6	0.186775	0.036102	0.001937	0.000067	0
0.7	0.214319	0.056189	0.004860	0.000238	0.000007
0.8	0.242463	0.078895	0.013960	0.001873	0.000301
0.9	0.265858	0.105294	0.032217	0.008317	0.002766
1.0	0.295408	0.140492	0.064803	0.033874	0.022211

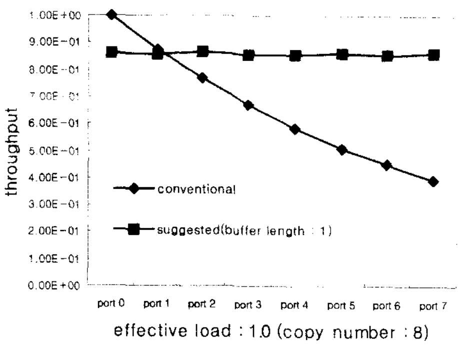


그림 10. 입력 포트간의 공평성
Fig. 10. Fairness among ports.

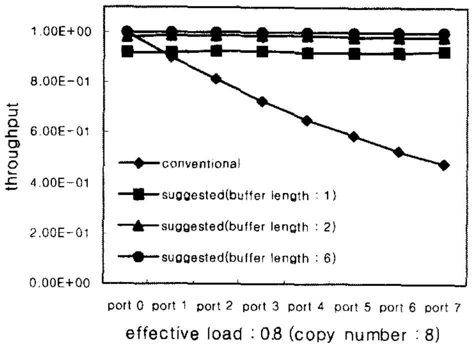
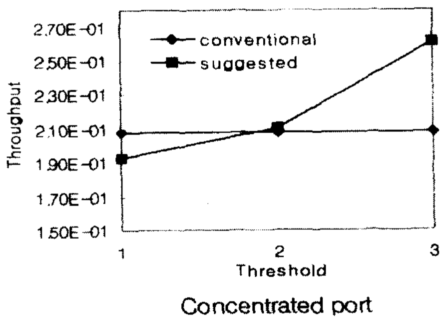
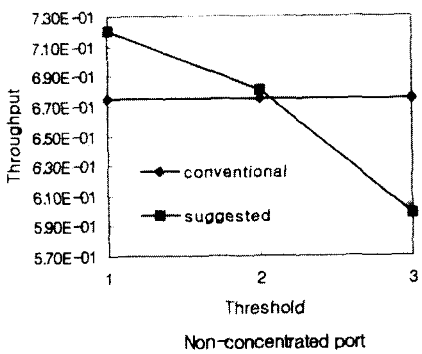


그림 11. 버퍼 길이에 대한 공평성
Fig. 11. Fairness versus buffer length.

Fig. 10 and 11 are the simulation result on the fairness between the conventional copy network [1] and the suggested copy network. As the figures indicate, the throughput in the conventional copy network decreases as the port number grows. This result is caused by the fixed summing scheme for the copy requests. On the other hand, the throughputs over all the input ports in the suggested copy network are almost even. Therefore, the fairness is improved in suggested copy network. The conventional copy network applying the *call splliting* algorithm [8] is suggested for the improvement of the throughput. Although it improves the throughput, it doesn't care the characteristics of the input traffics, so it doesn't adapt to the fluctuations of the input traffics dynamically. The simulation on the throughput for the bursty traffics is performed between the conventional copy network [8] and the suggested network. The simulation is performed under the condition that the buffer length is four for each copy network and input traffics concentrate on port 2 and 6. The probability that the packets arrive port 2 and 6 is 3/10 and it is 1/15(= 4/10*1/6) for each other port. Therefore, the traffic rate for the port 2 and 6 is about 4.5 times higher than that for others. It is shown from Fig. 12 that the throughput of the suggested copy network depends on the threshold of the buffer. That is, when the threshold is one, the throughputs of the suggested copy network for the concentrated ports are lower than that of conventional copy network. But, the situations are reversed as the threshold is passed one. This result is analyzed like this. That is, the input traffics result in the mean buffer length which is more than one but less than two for the non-concentrated buffers and more than or equal to two for the concentrated buffers. So, when the threshold is set to one, port 2 and 6 are not shown as the concentrated buffers to the suggested copy network. On the other hand, port 2 and 6 are considered as concentrated ports to the suggested copy network in case that the threshold is set to more than one. Therefore, the suggested copy network services the concentrated buffers more frequently compared to the conventional network. Therefore, it reduces the packet loss that is caused by the overflow in the concentrated buffers. This result is shown in Fig. 12(a). It is a reasonable result that more frequent service for the concentrated buffers causes less frequent service for the non-concentrated buffers. So, the throughput of the input ports whose occupancy is below the threshold is rather decreased in the suggested copy network in comparison with the conventional network. This is illustrated in Fig. 12(b). The same dependability of the copy network on the threshold is observed in Fig. 13. This is simulated under the conditions that buffer length is three and the other condition is same as Fig. 12. Therefore, it is concluded that the suggested copy network provides higher adaptability to the environments where input traffics are bursty and concentrated to some input



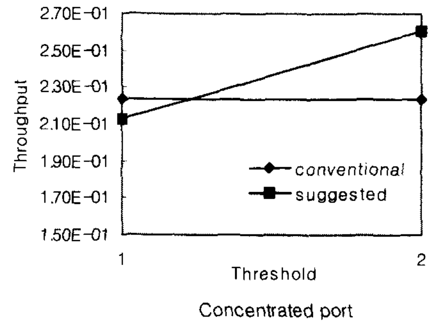
(a)



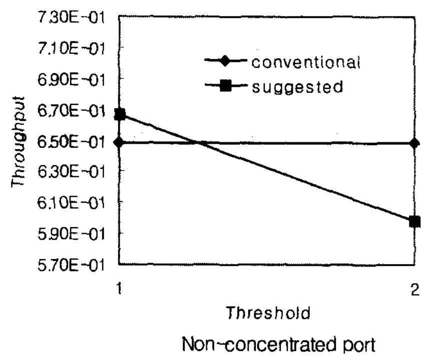
(b)

그림 12. (a) 집중 포트에 대한 스루풋 비교(버퍼 길이 : 4) (b) 비집중 포트에 대한 스루풋 비교(버퍼 길이 : 4)

Fig. 12. (a) The comparison of the throughput for the concentrated input ports(buffer length : 4) (b) The comparison of the throughput for the non-concentrated input ports(buffer length : 4).



(a)



(b)

그림 13. (a) 집중 포트에 대한 스루풋의 비교(버퍼 길이 : 3) (b) 비집중 포트에 대한 스루풋의 비교(버퍼 길이 : 3)

Fig. 13. (a) The comparison of the throughput for the concentrated input ports(buffer length : 3) (b) The comparison of the throughput for the non-concentrated input ports(buffer length : 3).

ports transiently as well as higher fairness than the conventional network.

Ⅳ. Conclusion

In this paper, a new copy network for the multicast switching is suggested and its performance is evaluated with a simulation. The suggested copy network features the adaptability to fluctuations of the input traffic and a dynamic change of the starting point for the summing of the copy requests. The hardware logic of the *DSD* is also derived with the objective of high speed. The derived logic is constructed with very simple structure. Therefore, it is suited to the high-speed environments. The consideration of the characteristics of input traffics in constructing copy networks results in better performance in loss rates than the conventional networks. Specifically, the suggested copy network only with 1 of buffer length improves the packet loss rate about 2 to 27 times over the conventional copy network. The improvement on the fairness among input ports is also very important result. *The throughput in the conventional copy network* decreases as the port number grows. On the other hand, the throughputs over input ports in the suggested copy network are almost even. This is accomplished with the dynamic change of starting points. In order to apply the copy network to the environments where the characteristics of the traffics are broad and bursty, the simple hardware structures for the *DSD* targeting high-speed operations should be devised and a timing skew for the feedback path which spans between *OFD* and *DSD* should be controlled for the reliable synchronized operation.

참 고 문 헌

[1] Tony T. Lee, "Non-blocking copy networks for multicast packet switching", *IEEE JSAC*. Vol. 6, pp. 1455-1467, Dec. 1988.

[2] Wen De Zhong, Yoshikuni Onozato, and Jaidev Kaniyil, "A copy network with shared buffers for large-scale multicast ATM switching", *IEEE/ACM Trans. On Networking*. Vol. 1, no.2 pp.157-165, Apr. 1993.

[3] Shigeo Urushidane, Shigeki Hino, Kimiyoshi Yamasaki, and Ken-ichi Yukimatsu, "A high-performance multicast switch for broadband ISDN", *XIV ISS '92*. Vol. 2, pp. 171-175, Oct. 1992.

[4] H.J.Chao, "A recursive modular terabit/second ATM switch", *IEEE JSAC*. Vol.9, no.8 pp. 1161-1171, Oct. 1991.

[5] David X. Chen and Jon W. Mark, "Multicasting in the SCOQ switch", *Proc. INFOCOM'94*. Vol.1 pp. 290-297, 1994.

[6] H. Jonathan Chao and Byeong Seog Choe, "Design and analysis of a large-scale multicast output buffered ATM switch", *IEEE Trans. On Networking*. Vol. 3, no. 2 pp. 126-138, Apr. 1995.

[7] H. Jonathan Cha, Byeong Seog Choe, Jin Soo Park, and Necdet Uzun, "Deisgn and inplementation of abacus switch: a scalable multicast ATM switch", *IEEE JSAC. Vol. 15*, no. 5 pp. 830-842, Jun. 1997.

[8] Jae W. Byun and Tony T. Lee, "The design and analysis of an ATM multicast switch with adaptive traffic controller", *IEEE/ACM Trans. On Networking*. Vol. 2, pp. 288-298, Jun. 1994.

[9] Xinyi Liu, H.T. Mouftah, "Queueing performance of copy networks with dynamic cell splitting for multicast ATM switching", IEEE Trans. On Comm. Vol. 45, no. 4 pp.464-471, Apr. 1997.

저 자 소 개

白 政 勳(正會員)

1982년 ~ 1986년 한양대학교 전자공학 학사. 1986년 ~ 1988년 한양대학교 전자공학 석사. 1988년 ~ 1996년 한국전자통신연구소. 1995년 ~ 현재 한양대학교 박사과정 수료, 현재 두원공과대학 전파통신과 교수

林 濟 鐸(正會員) 第 33卷 A編 第 7號 參照

현재 한양대학교 전자공학과 교수