크로스바 ATM 스위치에서의 장애 관리

오 민 석†

요 약

다중채널 스위치는 ATM (Asynchronous Transfer Mode)로 널리 사용되는 스위치 구조이며, 스위치의 내부에 장애에 대한 내성(tolerance)을 구현할 수 있는 것으로 알려져 있다. 예를 들어, 하나의 다중 채널 그룹에 속하는 링크에 장애가 있을 경우, 장애 링크로 통과하려는 트래픽을 여분의 링크가 책임을 질 수 있게 할 수 있다. 스위치 소자에 발생하는 장애는 ATM 셀을 잘못 라우팅하거나 출력단에 도달하는 셀의 순서를 뒤바꾸게 할 수 있다. 본 논문에서는 다중 채널 크로스바 ATM 스위치에 적용할 수 있는 두 가지의 장애 위치 확인 알고리즘을 제안한다. 첫 번째로 제안하는 최적 알고리즘은 시간적으로 최상의 성능을 보여주지만, 계산상으로는 복잡하게 되어 결과적으로 실제 구현이 어려울 수 있다. 이러한 문제점을 해결하기 위해 최적의 알고리즘보다는 계산상으로 보다 효율적인 온라인 알고리즘을 제안한다. 두 알고리즘의 성능은 시뮬레이션을 통해 검증한다. 온라인 알고리즘은 랜덤 트래픽 및 버스티(bursty) 트래픽에 대해 거의 최적에 가까운 성능을 보여 준다. 한편, 제안된 알고리즘으로 장애를 찾아낼 수 없는 경우가 있는데, 그에 대한 열거 및 원인을 제시한다. 끝으로 장애 위치 확인 알고리즘을 이용해서 찾은 장애를 우회하기 위해 행과 열을 추가하는 장애 복구 알고리즘을 제안하다.

Fault Management in Crossbar ATM Switches

Minseok Oh†

ABSTRACT

The multichannel switch is an architecture widely used for ATM (Asynchronous Transfer Mode). It is known that the fault tolerant characteristic can be incorporated into the multichannel crossbar switching fabric. For example, if a link belonging to a multichannel group fails, the remaining links can assume responsibility for some of the traffic on the failed link. On the other hand, if a fault occurs in a switching element, it can lead to erroneous routing and sequencing in the multichannel switch. We investigate several fault localization algorithm in multichannel crossbar ATM switches with a view to early fault recovery. The optimal algorithm gives the best performance in terms of time to localization but it is computationally complex which makes it difficult to implement. We develop an on-line algorithm which is computationally more efficient than the optimal one. We evaluate its performance through simulation. The simulation results show that the performance of the on-line algorithm is only slightly sub-optimal for both random and bursty traffic. There are cases where the proposed on-line algorithm cannot pinpoint down to a single fault. We enumerate those cases and investigate the causes. Finally, a fault recovery algorithm is described which utilizes the information provided by the fault localization algorithm. The fault recovery algorithm provides additional rows and columns to allow cells to detour the faulty element.

키워드 : 장애 위치 확인(Fault Localization), 장애 복구(Fault Recovery), 다중채널 스위치(Multichannel Switch), 크로스바 스위치(Crossbar Switch)

1. Introduction

Multichannel switches exploit the concept of *channel grouping* to provide higher performance (e.g., throughput, cell loss probability, delay) in ATM switches [1-8]. Instead of being routed to a specific output channel, a cell is routed to any channel belonging to an appropriate channel group. Very often, especially in the interior of the ATM network, it is sufficient to specify the path of the connection, not the specific channels within the path. This implies that a cell can be routed to any channel of a switch within a group of output channels, provided that it eventually leads to the correct end point via the same path.

As the demand for new applications soars, greater variability in bandwidth and traffic characteristics (e.g., session duration, burstiness) is expected. The advantages of statistically sharing a higher channel capacity under such conditions are well known [9]. For example, the link efficiency is increased as a result of the decreased

※ 본 연구는 2004년도 경기대학교 신진연구과제 지원에 의해 수행되었음.
† 정 회 원 : 경기대학교 전자공학부 교수
논문접수 : 2004년 9월 9일, 심사완료 : 2004년 11월 29일

burstiness of the incoming traffic. Bit pipes of higher rates are formed which allow a number of applications to share bandwidth by dynamically allocating time slots. A larger channel group size is less likely to incur blocking for a single ATM cell, for a burst of cells, or for a request for a new session. Similarly, other performance measures such as cell delay, probability of buffer overflow, and congestion improve when multiple channels are grouped together as a single resource.

Almost all implementations of fault-tolerant multistage interconnection networks (MINs) introduces redundancy in the network [15]. Most of these solutions are expensive in terms of number of extra switch modules per stage, and/or the size of the switching elements. These solutions have a high hardware complexity which needs complex and routing algorithms. Moreover, if resequencing of cells is needed at the end of the transmission of messages then it will be costly not only in terms of the additional logic or hardware needed but in terms of the additional delay to be incurred when doing the resequencing.

Beneš network, multiplane or parallel banyan network [16], and Itoh's network [17] are examples of fault-tolerant networks [18].

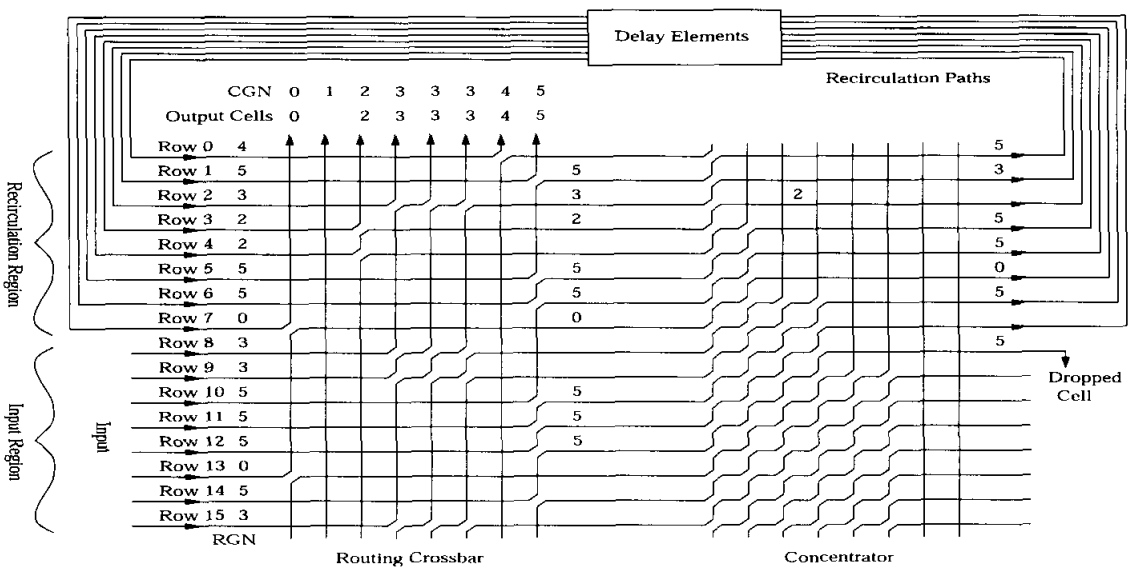
One of the important advantages of multichannel switches is the incorporation of inherent fault tolerance into the switching fabric [10]. For example, if a link which belongs to a multichannel group fails, the remaining links can assume responsibility for some of the traffic on the failed link. On the other hand, if a fault occurs in a switching element, it can lead to erroneous routing and sequencing in multichannel switches. We investigate this for multichannel architectures based on crossbars.

The problem we investigate is the rapid localization of faults in the switching element of a multichannel crossbar. The ability to localize such faults [11-13] rapidly allows for the incorporation of on-line fault recovery algorithms using redundant switching elements. Localization allows us to reduce the hardware overhead of extra switching elements by rerouting cells over only a small portion of the switch fabric.

In Section II we introduce the MCDC and MCOC architecture and the fault models we consider. In Section III we formulate an optimal time-to-detection fault localization algorithm and present simulation results. In Section IV, we describe a fault recovery algorithm which uses fault localization information obtained beforehand to route cells around a faulty switching element in the crossbar. Finally in Section V we describe an on-line algorithm and compare simulation results for it with those for the optimal algorithm. And we also investigate unlocalizable fault conditions and their causes.

2. MCDC and MCOC

Crossbar based switch modules provide space, power and clocking advantages [7, 14, 8] and integrate well with multichannel architectures. In particular, we use the *Multichannel Deflection Crossbar* (MCDC) and *the Multichannel One-turn Crossbar* (MCOC) architectures as our canonical architectures [7, 14].

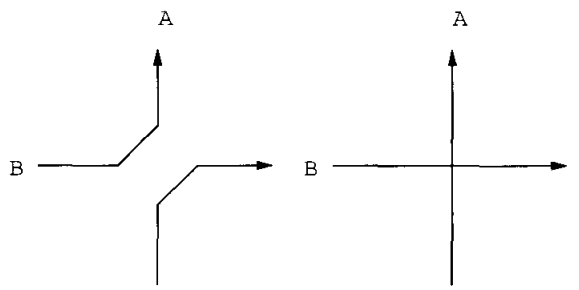


(Figure 1) Multichannel deflection crossbar architecture

2.1 MCDC

(Figure 1) shows an MCDC with 8 incoming inputs, 8 recirculating inputs and 8 output channels. It consists of a routing crossbar and a concentrator with recirculation paths connecting the output of the concentrator to the upper half of the input of the routing crossbar. Arriving cells enter the MCDC from the input channels located at the lower left portion of the routing crossbar and are destined for the output channels located at the top of the routing crossbar. We call the area in the routing crossbar including the rows in which arriving cells enter, the *input region*, and the area in the routing crossbar including the rows in which recirculating cells enter, the *recirculation region*, as indicated in (Figure 1). Each arriving cell has an output port address called *Requested Group Number* (RGN). This group number information is used to route the cell to the correct output port within the switch module. Each output channel is allocated a certain group number called the *Column Group Number* (CGN).

A crosspoint in the routing crossbar is a 2×2 switch element (SE). The main function of a 2×2 element is to determine the connection between two inputs (located to the left and below the crosspoint) and two outputs (located to the right and above the crosspoint). Each 2×2 element is set in one of the following two states during the switching operation as shown in (Figure 2).



(a) match and (b) bypass states

(Figure 2) States of a switching element

- *Match state*: This state corresponds to $A=B$, where is the group number of the vertical link (i.e., the CGN of the column) and B is the RGN of the cell appearing on the horizontal link. In this state the input from the left is connected to the output to the above and the input below the crosspoint is connected to the output to the right.
- *Bypass state*: This state corresponds to $A \neq B$. In this state the input from the left is connected to the output to the right and the input below the crosspoint is connected to the output to the above.

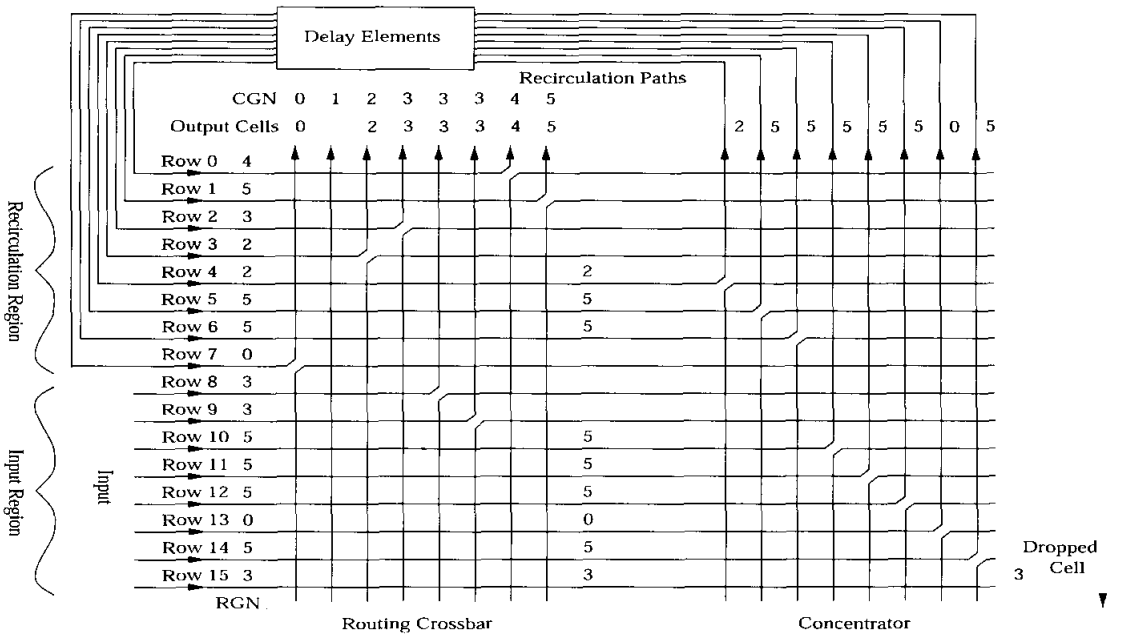
In each time slot, the states of 2×2 elements are computed first, and then the cells traverse along the path determined by the computed state of elements. Each cell moves horizontally rightward and when it encounters a 2×2 element in the match state, it starts to move vertically upwards. While a cell is traveling upwards, if a 2×2 element above is in the match state, it is deflected to the right, traveling horizontally rightward. The horizontal movement continues until another 2×2 element in the match state is encountered. If there is no other 2×2 element in the match state, the horizontal movement continues and the cell arrives at the input to the concentrator to be concentrated and recirculated.

The main function of the concentrator is to recirculate the cells that cannot exit from the routing crossbar in the present time slot to the input of the routing crossbar so that the cells can vie for the output again in the following time slot. The concentrator is a crossbar with the same number of horizontal rows as the routing crossbar as shown in (Figure 1). The state of a 2×2 element is determined by the logic operation, $A+\bar{B}$; A is the state of the 2×2 element directly above, i.e., $A=1$, if the element above is in the match state, and $A=0$ if it is in the bypass state. B indicates whether or not there is a cell arriving from the left at the crosspoint, i.e., $B=1$ if there is a cell, and $B=0$ otherwise. If the result of the operation $A+\bar{B}$ equals 1, then the 2×2 element goes into the match state. Otherwise, it is set to the bypass state. Note that a cell entering the concentrator can shift up at most by N rows if there are N columns in the concentrator (Think of the case when all the element it hits are set to the match state).

The recirculation paths provide shared buffering for the cells experiencing output contention. These cells are fed back to the input in order to vie for the output with the newly arriving cells in the next time slot. Delay elements are needed to provide separation between time slots.

2.2 MCOC

Another way of implementing the multichannel crossbar is to provide only one turn instead of several turns as in the MCDC. (Figure 3) illustrates the MCOC architecture. It also consists of a routing crossbar and a concentrator, but each element is configured differently from that in the MCDC. The MCOC requires a centralized controller which controls the state of each element; in the routing crossbar according to the CGN configuration and the RGN of cells in the left portion and in the concentrator according to the RGN of input cells to



(Figure 3) Multichannel one-turn crossbar architecture

the concentrator. The controller sets up the elements such that there is only one turn from an input port to an output port in the routing crossbar and in the concentrator (In the concentrator the input ports become the output ports of the routing crossbar and the output ports become the input ports to the shared buffer). For example, the first column of channel group whose group number is j is deflected only once at the intersection with the highest row among the rows whose RGNs are j. The second column of channel group j is deflected only once at the intersection with the second highest row among the rows whose RGNs are j. If there are more cells than available output channel in a slot, then the rows of the extra cells do not have any switch element configured as match state so that those extra cells can be routed to the concentrator for recirculation.

The concentrator can be considered as another routing crossbar, where the incoming cells to the concentrator are considered to have an identical RGN k and all the output ports are assigned with CGN k so that the incoming cells appear upward as compactly as possible without an empty row in the recirculation region in the following time slot.

We assume that each cell keeps its input port information while traveling along the path within the system to the output port so that the output port knows which input port the cell entered from.

2.3 Fault models

Throughout the investigation, we assume that the possible faults for the 2×2 SE are only either a *stuck-at match* (abbreviated as *s-a-m*) or *stuck-at-bypass* (abbreviated as *s-a-b*) fault. The *s-a-m* fault is a fault wherein the state of a 2×2 SE is permanently held in the match state and the *s-a-b* fault is a fault wherein the state of a 2×2 element is permanently held in the bypass state. In addition, we allow only for the possibility of a single fault. In reality, the probability of multiple faults is much smaller than that of a single fault justifying our assumption.

2.4 Notation

Before we proceed further let us define several variables for ease of explanation. N denotes the number of input ports for newly arriving cells in the input region and the number of output ports as well. We have R recirculation paths, i.e., shared buffers for the cells which could not exit through the output ports from the system in the previous slot. Then $M=N+R$, where M is the total number of rows in routing crossbar. Let K be the total number of channel groups in the system, and C_k be the *group capacity* of the kth channel group, $k=0, 1, \cdots, K-1$. The group capacity is defined as the number of output channels (columns) assigned to the same channel group. For example, in (Figure 1), $N=8$, $R=8$, $M=16$, $K=6$, $C_0=1$, $C_1=1$, $C_2=1$, $C_3=1$, $C_4=1$, and $C_5=1$.

3. Optimal Algorithm

In this section we develop an optimal localization al-

gorithm in terms of time to locate a faulty SE in the switch fabric. This algorithm is complex to implement and is investigated as the lower bound it provides for time to localize a failed element.

3.1 Algorithm

We consider only two types of faults in this fault localization scheme, *s-a-m* and *s-a-b* faults, as assumed earlier, which means that there are $2MN$ possible single faults in the $M\times N$ routing crossbar. However, due to the architecture and routing procedure in the MCDC and MCOC certain faults do not cause any deviation from normal operation(because either a switch element must always be in a particular state which coincides with the stuck-at fault for the element or because cells never reach the faulty element). They are called *undetectable* faults; let there be u of those. Hence we have hypotheses to test ($2MN-u$ cases of the localizable faults plus a case of the non-faulty case).

The optimal algorithm works as follows: First we assume $2MN-u+1$ imaginary switch systems along with an actual presumed faulty switch system (which is being tested). At the beginning of the algorithm we include all $2MN-u+1$ hypotheses in the *favored hypothesis set*. At each cell time slot, we store the input port information of output cells outcoming from the system, i.e., which rows they entered, and input port information of recirculating cells appearing in the following cell time slot, i.e., which cells appeared in what rows. In the first cell time slot we simulate the system using the same input cell and output channel configuration under $2MN-u+1$ different fault conditions. Then the output information obtained from the actual system which is being tested is compared with those from the $2MN-u+1$ imaginary systems through simulation. If they match, we leave the hypothesis in the favored hypothesis set. If not, we remove the hypothesis from the set. This process is repeated at each cell time slot. As time passes, the number of hypotheses left in the favored hypothesis set will be reduced and finally when there is one left in the set (this will happen with probability of one) the hypothesis will be the fault condition we are looking for and we stop the simulation.

Our algorithm is optimal in time for localization if the test for each hypothesis has the probability of one for correct detection. To see this, consider a hypothesis $\mathbf{H}_i$, $i=0, \cdots, 2MN-u+1$. Let the detectable fault associated with hypothesis $\mathbf{H}_i$ be f_i. Let $\mathbf{O}_j$ designate the input port information of output (outcoming from the system) cells in time slot j, i.e., which rows they entered, and input port information of recirculating cells appearing in the following cell time slot $j+1$, i.e., which cells appeared in what rows. What the algorithm does is to keep $\mathbf{H}_i$ in the favored hypothesis set if

$$P(f_i \mid \mathbf{O}_1, \mathbf{O}_2, \cdots, \mathbf{O}_T)>0 \tag{1}$$

and remove $\mathbf{H}_i$ from the set the first time when

$$P(f_i \mid \mathbf{O}_1, \mathbf{O}_2, \cdots, \mathbf{O}_T)=0 \tag{2}$$

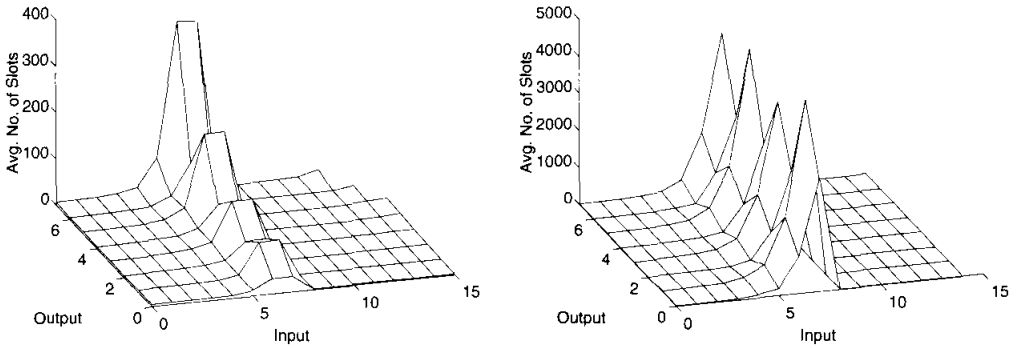
in the current time slot T. Suppose we remove $\mathbf{H}_i$ from the favored hypothesis set before Equation (2) is satisfied, i.e., while the outputs from simulation and the actual tested system match. Then the probability of correct detection will be less than one because hypothesis $\mathbf{H}_i$ may be true, which contradicts the given condition. Therefore the algorithm is optimal in time for detection.

3.2 Simulation Results of Optimal Algorithm

(Figure 4) shows the average numbers of required cell time slots until localization for *s-a-m* and *s-a-b* at each element in MCDC using the optimal algorithm. We used $M=16$, $N=8$ and the CGN assignment of (0, 0, 1, 1, 2, 2, 3, 3) at the output ports. It is assumed that the RGNs of the input cells were uniformly distributed in proportion to the group capacity of channel groups used in the system, i.e., in this example the probabilities of an incoming cell having RGN 0, 1, 2, and 3 are 1/4, 1/4, 1/4 and 1/4 respectively. It is also assumed that the traffic at one input port is independent of that at the other port. We used a traffic intensity of 0.7, i.e., 7 out of 10 slots are filled with cells on average.

First it is noticeable that the average numbers of slots for localization of *s-a-b* are greater than those for *s-a-m*. It is because localization of a *s-a-m* (*s-a-b*) requires a bypass (match) configuration on the faulty location (These opposite configurations cause cells to go in the other direction and result in different output cell combination from that in the non-faulty case). and the probability of match configurations in a slot is smaller than that of bypass configurations, which results in more cell time slots for localization of *s-a-b*.

We observe that the number of slots for localization is greater in the lower recirculation region, i.e., in input row 5, 6, and 7, than the other locations in both *s-a-m* and *s-a-b* cases. This is because when cells enter the concentrator they are shifted upward by at most N rows



(a) stuck at match (b) stuck at bypass

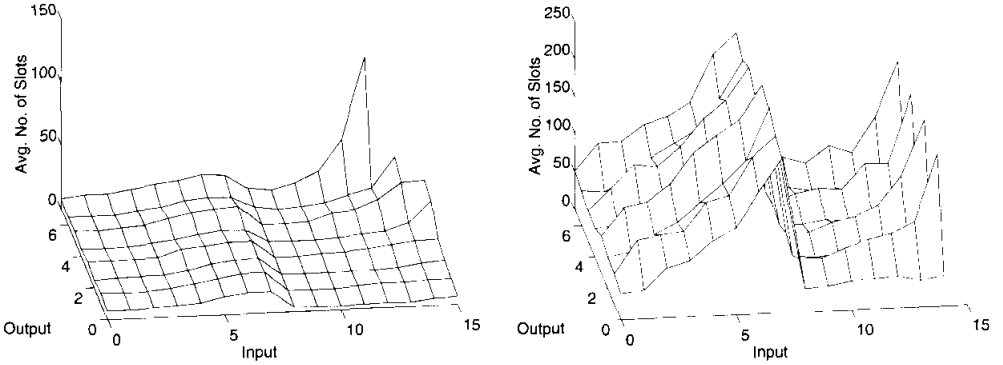
(Figure 4) No. of slots to localize in MCDC using the optimal algorithm

in MCDC and therefore it is difficult for cells to appear at the lower position among the input ports from the recirculation paths.

We also notice that in the case of *s-a-m* in (Figure 4) (a), the slot numbers for localization at the elements far away from the input ports are higher overall than those near the input ports, while in the case of *s-a-b* in (Figure 4) (b), the slot numbers for localization are even overall in any row in MCDC. This can be explained as follows: In order for a fault to be localized we need to have at least one mismatched output. To have the mismatched output, if the fault is a *s-a-m* (*s-a-b*) the corresponding element should be configured as bypass (match) state and the element should be reached by a cell. In the case of a s-a-m fault since a cell can exit through the output port from the system before it reach the faulty switch element, the element far away from the input ports will have less chance for cell to reach, which results in more slots for localization. But in the case of a *s-a-b* since only the cell which wants to deflect at the faulty location can help the localization process and the distribution of the deflecting cells is uniform due to the uniformity assumption of incoming cells' RGN, we have an even distribution of slots numbers for localization.

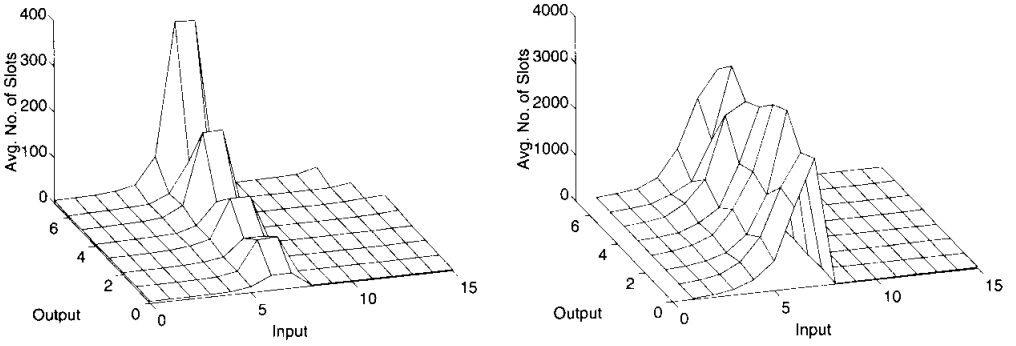
(Figure 5) shows the average number of slots for localization when the input traffic is bursty. We considered the cell arrivals to be ON-OFF sources, i.e., the cell arrivals alternate between the ON (arriving) and OFF (idle) states. Cell arrivals only occur in the ON state. The durations of ON and OFF periods are independent random variables exponentially distributed with means $1/\alpha$ and $1/\beta$ for ON and OFF periods, respectively. We define the burstiness as the mean value of the exponential distribution for state ON in the ON-OFF process, i.e., $1/\alpha$ and traffic intensity as β/α. We used burstiness of 5 and traffic intensity of 0.7 for simulation.

We see that the numbers of slots for localization in (Figure 5) are significantly lower compared to the case without burstiness in (Figure 4). This is because the burstiness of the input cells creates more recirculating cells and gives more chances for the cells to reach the lower recirculation region.



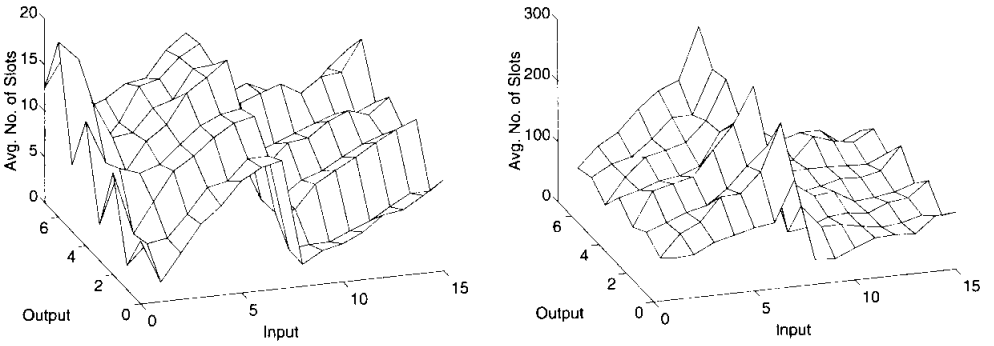
(a) stuck-at-match (b) stuck-at-bypass

(Figure 5) No. of slots to localize for bursty traffic in MCDC using the optimal algorithm



(a) stuck-at-match (b) stuck-at-bypass

(Figure 6) No. of slots to localize in MCOC using the optimal algorithm



(a) stuck-at-match (b) stuck-at-bypass

(Figure 7) No. of slots to localize for bursty traffic in MCOC using the optimal algorithm

(Figure 6) shows the average numbers of required cell time slots until localization for *s-a-m* and *s-a-b* at each element in MCOC using the optimal algorithm. We used the same condition as applied in the case of MCDC. We see that the slope of the number of slots for localization in the recirculation region is less steep in MCOC than the one in MCDC. This is because the concentrator in MCDC shifts cells upward at most by N even though there is an empty path above and hence the recirculating cells appear at the lower rows in the recirculation region, while the concentrator in MCOC shifts cells upward as compactly as possible and the recirculating cells always start to fill up rows from the top row in the recirculation region.

(Figure 7) shows the average numbers of required cell time slots until localization for *s-a-m* and *s-a-b* at each element in MCOC using the optimal algorithm for bursty traffic.

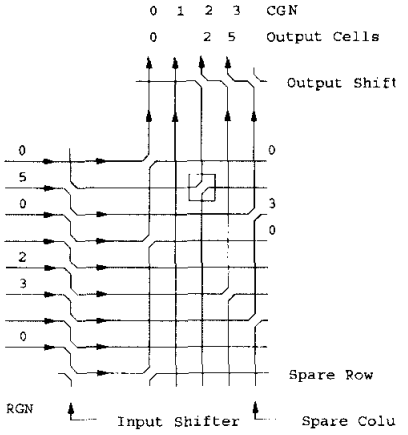
4. Fault Recovery Algorithm

Once a fault is detected and located, it is desired that the switch recovers from the fault.

Instead of duplicating the entire switching plane, we use shared rows and columns as backups for faulty SEs. Consider (Figure 8) where the routing crossbar is shown with two additional columns and two additional rows; one spare row and one spare column of SEs and one Input Shifter column and one Output Shifter row. The Input Shifter column and the Output Shifter row are used to make the path correction. Suppose that through the fault localization method proposed earlier, a fault on the 2×2 element located at the ith row and jth column has been detected. Then as shown in the figure, it is possible to eliminate the effect of the fault from the switch operation by reassignment of rows and columns by utilizing the additional rows and columns. In essence, this mechanism diverts the cells from the faulty paths to the unaffected capacity, utilizing all available capacity at its fullest, instead of passively accepting the degraded performance level.

In (Figure 8), two extra rows and two extra columns are needed to completely mask a single fault in the routing crossbar. Suppose the SE indicated with a box is faulty. Then, the input shifter shifts down all input ports to the

rows by one below the faulty element, thus bypassing the faulty SE and using the spare row in the process. The faulty column is not used and all columns are shifted to the right by one (thus using the spare column). The output shifter is used to shift the columns left by one in order to be aligned with the output channels.



(Figure 8) Fault recovery in Routing Crossbar

This scheme can be extended to masking k faults (Even though only a single fault is considered in the localization process). For example, it is possible to mask two faults by the addition of four extra rows (two rows for input shifter and two spare rows) and four extra columns (two columns for output shifter and two spare columns).

5. On-Line Localization

In this section, we develop an on-line fault localization scheme. The method is computationally more efficient than the optimal algorithm and its performance only slightly suboptimal. First, we present the basic concept upon which the proposed fault localization scheme is based. Then we describe the fault localization algorithm. Finally, we discuss some *incompletely localizable fault* conditions which cannot be localized to a single element.

5.1 Algorithm

We again utilize the entering row number information of incoming and recirculating cells. Once the RGNs of input cells at the input ports and the CGNs at the output ports are known, we can easily compute the exact path through which each cell will follow when there is no fault according to the switch configuration rules of MCDC or MCOC described earlier. Then we compare the correct output cells from the nonfaulty system with those from the system being tested to locate a possible faulty 2×2 element.

We define a s-a-m indicator matrix $\mathbf{H_m}$ and a s-a-b indicator matrix $\mathbf{H_b}$. The size of each indicator matrix is the same as the size of the routing crossbar, i.e., $M \times N$. Each element in $\mathbf{H_m}$ and $\mathbf{H_b}$ represents a *suspicion* value which is a measure of the current suspiciousness of the corresponding element with regard to s-a-m and s-a-b respectively. That is, the bigger the value in $\mathbf{H_m}$ and $\mathbf{H_b}$ is, the more suspicious the element is with regard to s-a-m and s-a-b respectively.

The suspicion value is set to 0 initially. As the localization process proceeds, the suspicion value increases by one in a cell time slot, whenever the outputs from the nonfaulty and tested systems do not match and the corresponding element lies on the correct paths of the mismatched outcoming cells. Here the correct path means the path which the cell would have taken if there had not been a fault. Consequently, after enough cell time slots have passed, the unchanged value 0 in $\mathbf{H_m}$ and $\mathbf{H_b}$ means that the corresponding 2×2 element is not s-a-m and not s-a-b, respectively, at the moment.

Since we already know the RGNs of cells entering input ports and the CGNs at the output channels, we can compute the paths of entering cells and predict what cells will come out of the output ports. As mentioned earlier, cells traveling in MCDC will carry their own input port number information, i.e., the row numbers which they entered. We can compare the row (port) numbers of the cells outcoming from MCDC being tested with the row (port) numbers outcoming from the nonfaulty MCDC. If those row numbers are not the same, we know there is at least one faulty element on the correct path of the routed cell and therefore increase the corresponding suspicion values in either $\mathbf{H_m}$ or $\mathbf{H_b}$ according to the following rule: If the element on the correct path was configured as the bypass state, we increase the value in $\mathbf{H_m}$ and if configured as the match state, we increase the value in $\mathbf{H_b}$. If those input port numbers from both non-faulty and tested systems are the same, we judge that there is no fault on the path of the routed cell and then clear suspicion on all the elements lying on the path by setting the suspicion value to 0. If the element on the correct path was configured as bypass state, we set the suspicion value of $\mathbf{H_m}$ to 0 and if configured as match state, we set the suspicion value of $\mathbf{H_b}$ to 0. These elements set to 0 is never increased later because those elements are believed to be nonfaulty. We continue to do this until we get a unique maximum element in $\mathbf{H_m}$ and $\mathbf{H_b}$.

Proposition: If the maximum of the suspicion values in $\mathbf{H_m}$ and $\mathbf{H_b}$ is achieved by a unique element, then the element associated with the unique maximum is the faulty element.

Proof: The suspicion values on the correct path increase when a cell is misrouted due to a fault on the correct path. Suppose the actual fault is a *s-a-m* fault at a localizable location A (i.e., a fault which can be localized by the algorithm above; the condition for localizability will described shortly) and we have the unique maximum at location B in any of $\mathbf{H_m}$ and $\mathbf{H_b}$ which is different from location A in $\mathbf{H_m}$. Let the suspicion value of $\mathbf{H_m}$ at A be S_A and that at B be S_B. By the given hypothesis, $S_A < S_B$. Since the mismatched outputs are solely due to the *s-a-m* fault at location A, whenever we have mismatched outputs the algorithm always increase the suspicion value at location A in $\mathbf{H_m}$. Therefore whenever there is an increase in either $\mathbf{H_m}$ or $\mathbf{H_b}$, the suspicion value at A in $\mathbf{H_m}$ will be increased. Therefore $S_A \geq S_B$, i.e., we have a contradiction. This proves the proposition for the case of *s-a-m*. The argument is true when we have a *s a-b* fault as well. **Q.E.D.**

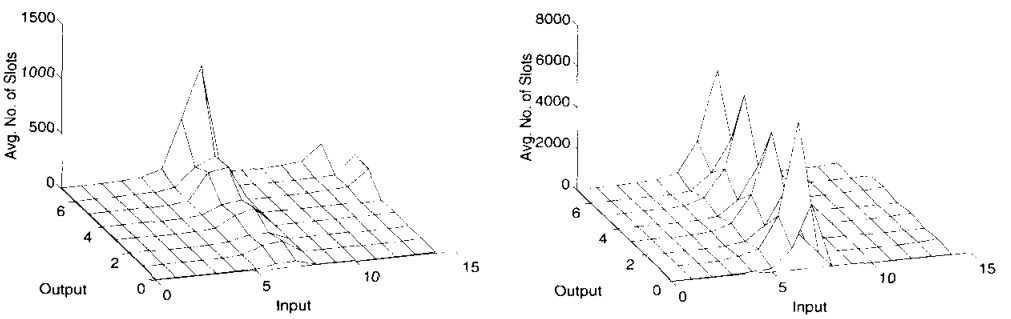
The algorithm is repeated until there exists a unique maximum among elements in either $\mathbf{H_m}$ or $\mathbf{H_b}$. After the algorithm is repeated long enough, most of the values become zero because most elements are marked as free of *s-a-m* or free of *s-a-b* when paths are confirmed as non-faulty.

5.2 Simulation Results of On-Line Localization algorithm

We applied the same condition as in the case of the optimal case. That is, we applied the on-line algorithm for MCDC with $M=16$, $N=8$, and the CGN assignment of (0, 0, 1, 1, 2, 2, 3, 3) at the output ports. We assume that the RGNs of the input cells are uniformly distributed in proportion to the capacities of the CGNs with traffic intensity of 0.7 and the traffic at one input port are independent of the traffic at the other port.

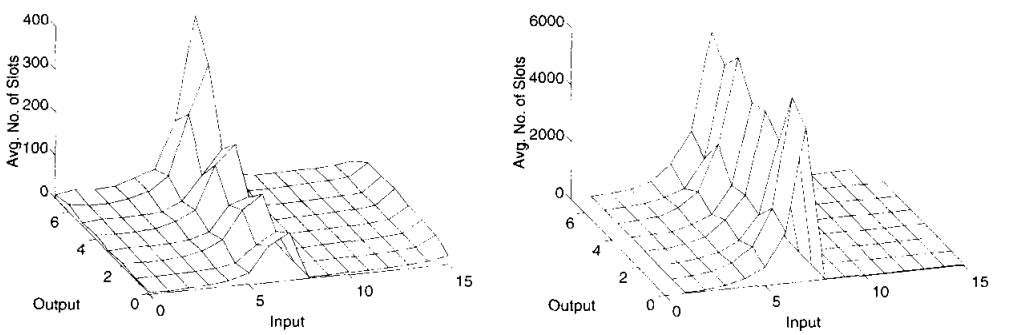
(Figure 9) and (Figure 10) shows the number of cell time slots until localization of a single fault at the corresponding location on the average over 100 simulations for *s-a m* and *s-a b* faults in the MCDC and MCOC, respectively. In the case of the MCOC, the output graphs show a similar pattern.

We also used bursty traffic as input. (Figure 11) shows the number of slots for localization for bursty input in the MCDC.



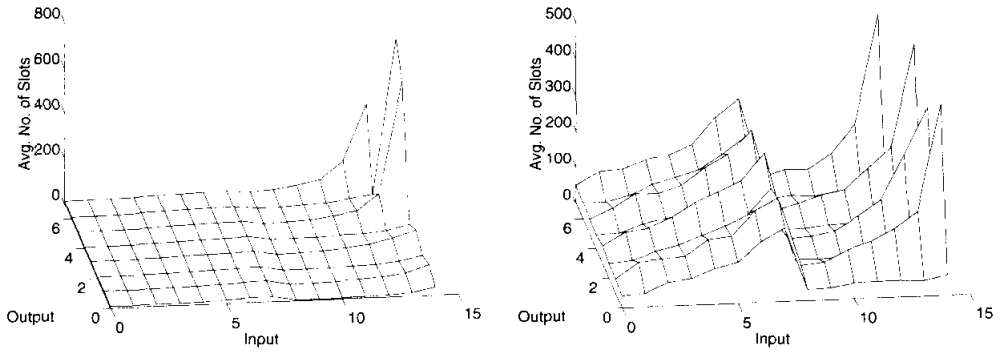
(a) stuck-at-match (b) stuck-at-bypass

(Figure 9) No. of slots to localize in MCDC using on-line algorithm



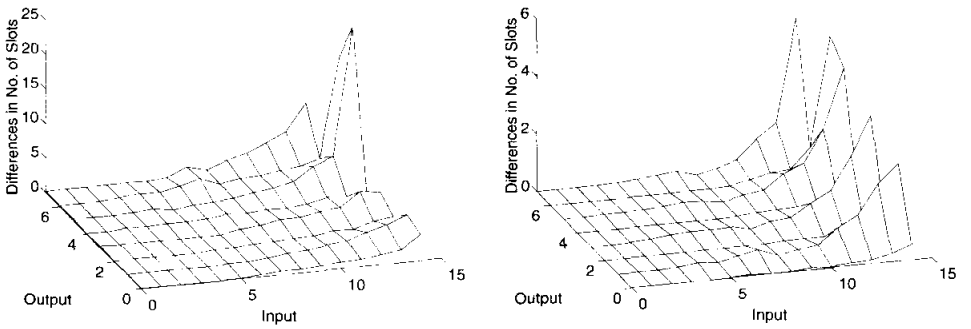
(a) stuck-at-match (b) stuck-at-bypass

(Figure 10) No. of slots to localize in MCOC using the on-line algorithm



(a) stuck-at-match (b) stuck-at-bypass

(Figure 11) No. of slots to localize in MCDC for bursty inputs with B=5 using the on-line algorithm



(a) stuck-at-match (b) stuck-at-bypass

(Figure 12) Difference in no. of slots to localize in MCDC between the optimal and on-line algorithms

(Figure 12) shows the differences between the average number of slots for localization using **H** matrix algorithm and using the optimal algorithm at a certain location. When the average number of slots for localization using **H** matrix algorithm is n_{H} and the average number of slots for localization using the optimal algorithm is n_{O}, the number in the graphs represents $(n_{\mathrm{H}}-n_{\mathrm{O}})/n_{\mathrm{O}}$. The plot shows that the number of slots for localization is close to the optimal one in most of the locations.

5.3 Incompletely Localizable Faults

There are cases where the proposed algorithm cannot pinpoint down to a single fault. This occurs when there is no unique element with the maximum value among the elements either $\mathbf{H}_{\mathrm{m}}$ or $\mathbf{H}_{\mathrm{b}}$. In some cases cells either never go through the faulty element, or the faulty element which is stuck at a certain state always needs to be configured as the faulty state (either *s-a-m* or *s-a-b*). In those cases, suspicion values never increase in either $\mathbf{H}_{\mathrm{m}}$ or $\mathbf{H}_{\mathrm{b}}$.

Those *undetectable* do not cause any problem to normal operation as long as the CGN configuration remains unchanged. In what follows we classify all the incompletely localizable faults *s-a-m* and *s-a-b* in the MCDC.

(1) Unlocalizable Stuck-At-Match Faults in MCDC: Unlocalizable *s-a-m* faults in the MCDC must belong to one of the five categories described below.

a. Consider a channel group other than the rightmost one in the routing crossbar. When the group capacity of the channel group is C_k (k=0, ···, K−1, K is the total number of channel groups), where C_k is greater than 1 and the channel group starts at column j, unlocalizable *s-a-m* faults are positioned at the following locations.

$$\begin{array}{cccc} & & (A+1,B-2) & (A+1,B-1) \\ & (A+2,B-3) & (A+2,B-2) & (A+2,B-1) \\ \iddots & \vdots & \vdots & \vdots \\ (M-1,j) & \cdots\ (M-1,B-3) & (M-1,B-2) & (M-1,B-1) \end{array}$$

where $A=M-C_k$ and $B=j+C_k$. In (Figure 13), the solid slashes indicate the unlocalizable single *s-a-m* faults under the given output group channel configurations in a 16×8 MCDC. We can see clearly the above rule for unlocalizable *s-a-m* fault holds except for (13, 6) in (b) and (15, 4) in (c), where (i, j) indicates the location at row i (i=0, ···, M−1) and column j (j=0, ···, N−1). Those exceptional

cases are explained in (b) and (c) respectively.

Reason: In order for a *s-a-m* fault to be localized a cell must try to pass straight through the faulty element either vertically or horizontally. But a cell can never pass straight vertically through the marked locations due to the output channel assignment (unless we have more rows below row 15.) Therefore we have to rely on a cell passing straight horizontally through the faulty element for localization. For example, consider a *s-a-m* at (15, 3) in (Figure 13)(a). The only way to send a cell horizontally through (15, 3) in (a) will be to insert a cell having RGN 2 in row 15. Then the proposed algorithm increases the suspicion values at (15, 3) and (15, 4) in $\mathbf{H}_m$ along with some other locations if the cell can exit from the system through the output port in the current cell time slot and result in a mismatch output. But the element at (15, 4) will never have a chance to be set as free of *s-a-m*, because any cell which wants to pass straight horizontally through (15, 4) will be deflected at (15, 3) due to the fault before it reaches (15, 4). Therefore there will be no unique maximum element throughout $\mathbf{H}_m$ and $\mathbf{H}_b$. Note that in the proposed algorithm in order for an element to be set as free of *s-a-m* (*s-a-b*) a cell should pass through the element configured as bypass(match) state and exit from the system through the output port in the same slot. When a *s-a-m* fault is located at (15, 4) we need a cell passing straight horizontally through the fault, say a cell with RGN 2 in row 15, and then it will increase the suspicion value at (15, 3) and (15, 4) in $\mathbf{H}_m$. And the element at (15, 3) will never be set as free of *s-a-m* because any cell which wants to pass straight horizontally through (15, 3) will be deflected at (15, 4) due to the fault and the algorithm will increase the values at (15, 3) and (15, 4) in $\mathbf{H}_m$, which results in no unique maximum. The unlocalizable faults at the other locations can be explained by the same argument.

b. When the rightmost channel group has a group capacity $C_{K-1}=1$ and the second rightmost channel group has a group capacity C_{K-2}, then the *s-a-m* fault at ($M-C_{K-2}$, $N-2$) is unlocalizable. (Figure 13)(b) shows the same kind of unlocalizable *s-a-m* at (13, 6).

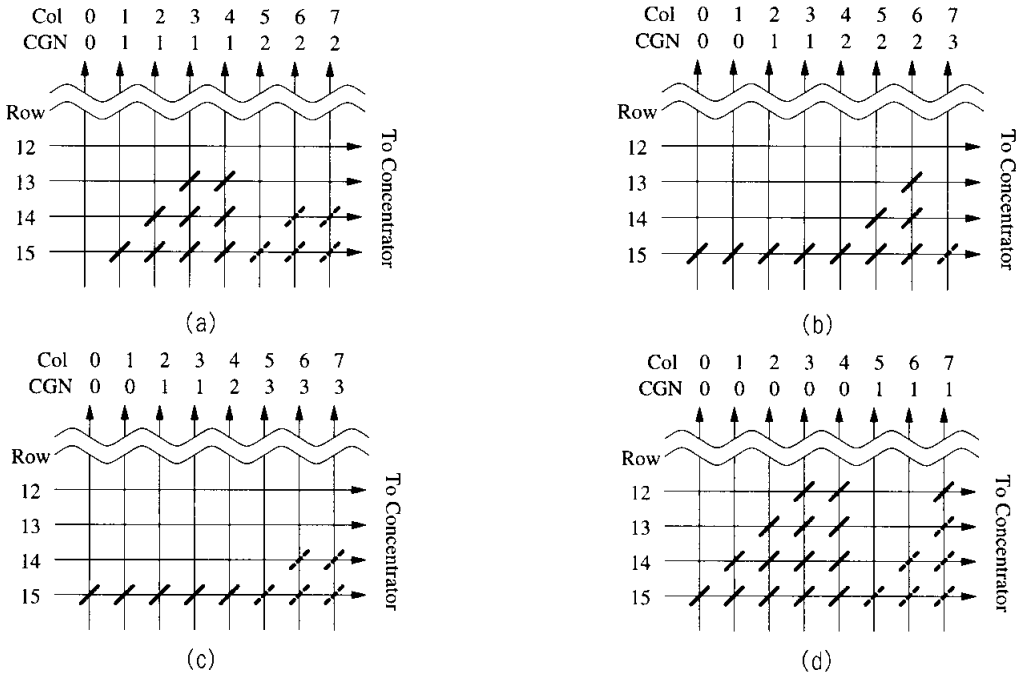
Reason: In order for the *s-a-m* at (13, 6) to be localized, we need a cell having RGN 3 at row 13. The cell passes straight through (13, 6) and always deflects at (13, 7) so that the suspicion value at (13, 6) in $\mathbf{H}_m$ and the value at (13, 7) in $\mathbf{H}_b$ always increase together. In order for the suspicion value at (13, 7) in $\mathbf{H}_b$ to be cleared, a cell should enter (13, 7) either through the left link or through the lower link, deflect and exit from the system through the output port in the same cell time slot. But the cell entering through the left link never exists due to the *s-a-m* at (13, 6) on the left of (13, 7) and the cell entering through the lower link and deflecting rightward ((13, 7) should be configured as *s-a-m* in order for *s-a-b* to be cleared.) will never come out of the system through the output port in the current slot. Therefore the suspicion value at (13, 7) will never be cleared, which makes the *s-a-m* at (13, 6) unlocalizable.

c. When the second rightmost channel group has a group capacity $C_{K-2}=1$, despite of its group capacity of 1, the *s-a-m* at the lowest row on the second rightmost channel group is unlocalizable. (Figure 13)(c) shows that a *s-a-m* at (15, 4) is unlocalizable.

Reason: We want a cell to pass straight through the *s-a-m* element in order to localize it. The only way for a cell to pass straight through the faulty element at (15, 4) is to have a cell having RGN 3 at row 15. In the nonfaulty system the cell will pass straight through (15, 4) and always deflect at (15, 5). Therefore (15, 4) in $\mathbf{H}_m$ and (15, 5) in $\mathbf{H}_b$ always increase together when we have a mismatching output. In order to clear the value at (15, 5) in $\mathbf{H}_b$, we have to insert a cell with RGN 3 in row 15, but the cell never reaches (15, 5) due to the fault at (15, 4)

d. When the group capacity of the rightmost channel group C_{K-1} is smaller than that of any other channel group, then a *s-a-m* at ($M-C_{K-1}-1$, $N-1$) is unlocalizable. In (Figure 13)(d), the fault at (12, 7) is unlocalizable because the rightmost channel group has the smallest channel group capacity in the system.

Reason: In order to detect the *s-a-m* at (12, 7) in (Figure 13)(d), we need a cell to pass straight through (12, 7) either horizontally or vertically. But a cell cannot pass straight horizontally through (12, 7) since row 10 is the lowest row in the rightmost channel group region which a cell can travel horizontally. Note that when the smallest group capacity of the channel groups to the left of the rightmost channel group is C_m, then the lowest row a cell can pass straight horizontally through the rightmost column is



(Figure 13) Unlocalizable s-a-m faults in MCDC(solid: unlocalizable, dotted: undetectable)

M-C_m-1. Therefore we need a cell to pass straight through (12, 7) vertically. It is possible only when the incoming inputs have RGNs ($\bar{1}$,$\bar{1}$, ··· ,$\bar{1}$,1,1,1), where the over-line indicates any group number which is not equal to the number under the over-line. (In (Figure 13)(d), $\bar{1}$ becomes 0.) But it increases the suspicion values at (12, 7) in $\mathbf{H_m}$ and at (13, 7) in $\mathbf{H_b}$ together. In order to clear the value at (15, 5) in $\mathbf{H_b}$, we need a cell deflecting at (13, 7) either horizontally or vertically. But a cell entering (13, 7) vertically does not exist due to the CGN configuration and a cell entering horizontally never comes out of the system due to the *s-a-m* fault at (12, 7)

e. When the rightmost channel group has a group capacity C_{K-1} and no channel group to the left of the rightmost one has a group capacity less than C_{K-1}, the single *s-a-m* faults at the following locations are undetectable *s-a-m* faults. Again, an undetectable fault is defined as a fault which does not affect normal switching operation and is not detected.

$$\begin{array}{cccc} & & & (A, N-1) \\ & & (A+1, N-2) & (A+1, N-1) \\ & \iddots & \vdots & \vdots \\ (M-2, B+ & \cdots & (M-2, N-2) & (M-2, N-1) \\ (M-1, B) & (M-1, B+ & \cdots (M-1, N-2) & (M-1, N-1) \end{array}$$

where $A=M-C_{K-1}$ and $B=N-C_{K-1}$. In (Figure 13)(a), (b), (c), and (d), all the dotted slashes indicate undetectable *s-a-m* fault locations. However, if any channel group to the left of the rightmost one has a group capacity less than C_{K-1}, then the *s-a-m* fault at ($M-C_{K-1}$, N-1) located in the upper right position of the above list becomes localizable. The *s-a-m*'s at (13, 7) in (Figure 13)(a) and (c) become localizable due to the same reason.

Reason: If no channel group to the left of the rightmost one has a group capacity less than C_{K-1}, then no cell will reach the above locations except for those diagonal elements at ($M-C_{K-1}$, N-1), ($M-C_{K-1}+1$, N-2), ···, and (M-1, $N-C_{K-1}$). Even those diagonal elements are always configured as the match state when there is an input cell entering the corresponding rows. Therefore all the faults in the above locations cannot affect normal operation.

(2) Unlocalizable Stuck-At-Bypass Faults in MCDC: Unlocalizable *s-a-b* faults in the MCDC must belong to one of the following three categories.

a. When the rightmost channel group has a group capacity C_{K-1}, the *s-a-b* at ($M-C_{K-1}$, N-1) is unlocalizable. (13, 7) in (Figure 14) illustrates the corresponding unlocalizable *s-a-b* fault.

Reason: To localize the *s-a-b* at (13, 7) we need cells with RGN 2 at row 13, 14, and 15. Then in the nonfaulty system the cell entering row 15 will reach (13, 7). Then the cell deflects at (13, 7) passes straight through (12, 7) and comes out of the

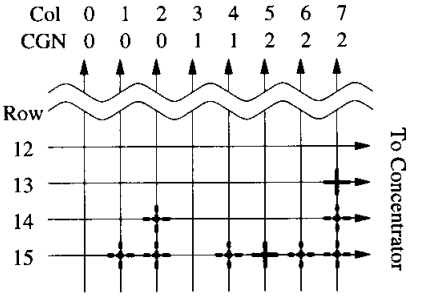
system through the output port if no row has a input cell having RGN 2 above row 13. In the faulty system the cell cannot come out through the output port due to the *s-a-b* fault at (13, 7). Therefore the suspicion values at (13, 7) in $\mathbf{H}_h$ and at (12, 7) in $\mathbf{H}_m$ always increase together, along with some other location. We need a cell passing straight through (12, 7) and exiting from the system through the output port in the same cell time slot, in order to clear the suspicion value at (12, 7) in $\mathbf{H}_m$. Since a cell passing straight through (12, 7) horizontally does not come of the system through the output port in the current time slot, the case never happens. Moreover, a cell passing straight through (12, 7) vertically does not exist due to the *s-a-b* at (13, 7). (Note that a cell passing straight through (13, 7) vertically does not even exist under this channel group configuration.) Therefore we cannot clear the suspicion value at (12, 7) and there will be no unique maximum in both suspicion matrices.

b. When the rightmost channel group has a group capacity C_{K-1}, the *s-a-b* at $(M-1, N-C_{K-1})$ is unlocalizable. (15, 5) in (Figure 14) illustrates the corresponding unlocalizable *s-a-b* fault.

Reason: To localize the *s-a-b* at (15, 5), we need a cell having RGN 2 at row 15 so that the cell can deflect at (15, 5). Then the proposed algorithm increases the suspicion values at (15, 3) and (15, 4) in $\mathbf{H}_m$ and (15, 5) in $\mathbf{H}_h$ along with some other locations. In order to clear the suspicion values at (15, 3) and (15, 4) in $\mathbf{H}_m$ we need a cell passing straight through those elements and coming out of the system in the same time slot. The only way for a cell to pass straight through those elements is to have a cell with RGN 2 at row 15, but the cell never comes out of the system due to the *s-a-b* at (15, 5). Therefore there is no way to clear those two values

c. When a channel group has a group capacity C_K and the first column of the channel group starts at column j, the single *s-a-b* faults at the following locations are undetectable.

$$\begin{array}{cccc} & & & (A+1,B-1) \\ & & (A+2,B-2) & (A+2,B-1) \\ & & \vdots & \vdots \\ (M-2,j+2) & \cdots & (M-2,N-2) & (M-2,B-1) \\ (M-1,j+1) & (M-1,j+2) & \cdots & (M-1,B-2) \quad (M-1,B-1) \end{array}$$



(Figure 14) Unlocalizable *s-a-b* faults in MCDC(solid: unlocalizable, dotted: undetectable)

where $A=M-C_K$ and $B=j+C_K$. In (Figure 14), all the dotted crosses indicate undetectable single *s-a-b* fault locations.

Reason: All locations marked with a dotted line cross are always passed straight through horizontally by cells except for the rightmost channel group region so that the *s-a-b* at the location other than the rightmost channel group region does not affect normal operation. The reason that the locations with the dotted crosses in the rightmost channel group region are not detected is because no cell passes the locations.

6. Conclusions

The problem we investigate is the rapid localization of faults in the switching element of a multichannel crossbar. The ability to localize faults rapidly allows for incorporation of on line fault recovery algorithms using redundant switching elements. Localization allows us to reduce the hardware overhead of extra switching elements by rerouting cells over only a small portion of the switch fabric.

We have investigated two localization algorithms including the optimal one in the canonical multichannel crossbar switches, i.e., Multichannel Deflection Crossbar (MCDC) and Multichannel One-turn Crossbar (MCOC) for the fault types of a stuck-at-match and a stuck-at bypass.

The optimal algorithm gives the best performance in terms of time to localization but suffers from computational complexity which makes it difficult to implement. The proposed algorithm is computationally more efficient than the optimal algorithm. The simulation results indicate that a *s-a-b* requires more time for localization than a *s-a-m* on average. It is because localization of a *s-a-m* (*s-a-b*) requires a bypass (match) configuration on the faulty location and the probability of match con-

figurations in a slot is smaller than that of bypass configurations, which results in more cell time slots for localization of s-a b. The simulation also results show that the bursty input traffic reduces the time to localization. It is because the burstiness helps cells reach the recirculation region which is hard to be reached by nonbursty traffic. A fault recovery algorithm has been described, which provides additional rows and columns to go around the faulty element. The on-line algorithm is computationally more efficient than the optimal algorithm and its performance only slightly sub-optimal. If the computational complexity of the on line algorithm is considered to be still unrealistic, the outgoing cell information can be stored for a period of time and the faulty element can be localized using the information later. Several locations in the routing crossbar are un localizable due to the internal routing algorithm and the localization characteristics, but some of them do not affect the normal switching operation.

References

[1] A. Pattavina, "Multichannel bandwidth allocation in a broadband packet switch," IEEE Journal on Selected Areas in Communications, Vol. 6, No.9, pp.1489 - 1499, Dec., 1988.

[2] R. L. Cruz, "The statistical data forkA class of broad-band multichannel switches," IEEE Transactions on Computers, Vol.40, No.10, pp.1625 - 1634, Oct., 1992.

[3] H. S. Kim, "Multichannel ATM switch with preserved packet sequence," IEEE International Conference on Communications, Vol.3, pp.1634 - 1638, 1992.

[4] A. Y.-M. Lin and J. A. Silvester, "On the performance of an ATM switch with multichannel transmission groups," IEEE Transactions on Communications, Vol.41, No.5, pp.760 - 770, May, 1993.

[5] P. S. Min, H. Saidi, and M. V. Hegde, "Nonblocking architecture for broadband multi-channel switching," IEEE/ACM Transactions on Networking, Vol.3, No.2, pp.181 - 198, 1995.

[6] T.-H. Cheng, "Design and analysis of a multichannel transmission scheme," Computer Networks and ISDN Systems, Vol.29, No.2, pp. 209 - 220, Jan., 1997.

[7] P. Y. Yan, K. S. Kim, P. S. Min, and M. V. Hegde, "Multi-channel deflection crossbar (MCDC)A VLSI optimized architecture for multichannel ATM switching," in Proceedings of INFOCOM '97, Kobe, Japan, pp.12 - 19, Apr., 1997.

[8] K.-B. Kim, P. Y. Yan, K. S. Kim, O. Schmid, and P. S. Min, "A growable ATM switch with embedded multichannel multicasting property," in IEEE GLOBECOM, pp.222 - 226, Nov., 1997.

[9] D. Bertsekas and R. Gallager, Data Networks, 2nd ed. Prentice Hall, 1992.

[10] T. Anderson, Fault TolerancePrinciple and Practice. Prentice Hall, 1981.

[11] A. T. Bouloutas, S. Calo, and A. Finkel, "Alarm correlation and fault identification in communication network," IEEE Transactions on Communications, Vol.42, pp.523 - 533, 1994.

[12] I. Katzela and M. Schwartz, "Schemes for fault identification in communication networks," IEEE/ACM Transactions on Networking, Vol.3, pp.753 - 764, 1995.

[13] A. A. Lazar, W. Wang, and R. H. Deng, "Models and algorithms for network fault detection and identificationA review," Communications on the Move. ICCS/ISITA '92, Vol.3, 1992, pp.999 - 1003.

[14] P. Y. Yan, "Crossbar architectures for broadband switching," D.Sc., Washington University, St. Louis, MO, 1997.

[15] S. K. Hui, K. Seman, and J. Yunus, "An augmented chained fault-tolerant ATM switch," 5th IEEE International Conference on High Speed Networks and Multimedia Communications, pp.397-400, 2002.

[16] J. T. Blake and K. S. Trivedi, "Multistage interconnection network reliability," IEEE Transactions on Computers, Vol.38, No.11, pp.1600 1604, Nov., 1989.

[17] A. Itoh, "A fault-tolerant switching network for B-ISDN," IEEE Journal on Selected Areas in Communications, Vol.9, No.8, pp.1218-1226, Oct., 1991.

[18] M. Anan and M. Guizani, "A fault tolerant ATM switching architecture," IEEE International Conference on Performance, Computing, and Communications Conference, pp.295-301, 2000.

오 민 석

e-mail : msoh@kyonggi.ac.kr

1987년 서울대학교 전기공학과

1993년 Columbia University 전기공학과 석사

1998년 Washington University 전기공학과 박사

1998년~1999년 minMax Technology 연구원

1999년~2000년 AT&T Technical Consultant

2000년~2004년 LG TeleCom 부장

2004년~현재 경기대학교 전자공학부 전임강사

관심분야 : 망관리, 이동통신