

# 서브 스위치를 이용한 오류 허용 정렬 네트워크

김흥진\* 손유익\*\*

계명대학교 컴퓨터공학전공

e-mail : khj0011@jinri.kmu.ac.kr yeson@kmucc.kmu.ac.kr

## Fault-tolerant sorting network with sub-switches

Heung-Jin Kim\*, Yoo-Ek Son\*\*

Department of Computer Eng. Keimyung University

### 요 약

본 연구에서는 서브 스위치를 이용한 오류허용 정렬 네트워크를 제안한다. 기존의 제안된 정렬 네트워크에서는 여분의 경로와 네트워크의 복잡성문제가 있었다. 제안된 구조에서는 여분 경로를 확장시키기 위해서 각 스테이지마다  $\frac{N}{4} \sum_{i=1}^2 (\frac{1}{2})^i$  (N=입·출력수) 서브 스위치를 추가함으로써 여분의 경로가  $3^{*(n+1)/2}$  만큼 증가하였다. 또한 제안된 정렬 네트워크는 기존의 네트워크의 이중 네트워크 plane 개념에서 사용한 스위치 소자와 링크 수와 비교해 볼 때 제안된 구조의 단일 plane을 이용한 구조가 복잡도에서 낮다. 결론적으로 제안된 서브 스위치를 이용한 오류 허용 정렬 네트워크는 여분의 경로를 증가시키면서 하드웨어적 복잡도를 감소시킬 수 있다.

### 1. 서론

정보화사회의 발전에 따라 원격화상회의, 음성전달, 비디오, 분산 데이터 처리 등과 같은 광범위한 서비스 요구 증대에 따라 B-ISDN과 함께 ATM 스위칭 기법에 대한 사용이 주목받고 있다. ATM은 큰 유연성과 함께 음성전송과 높은 속도의 데이터를 낮은 속도의 데이터로 동영상의 분산을 제어할 수 있다. 이러한 특징은 B-ISDN의 요구사항 중에서 다른 소스로부터 셀을 받아들여 목적지로 전송하는 기술을 잘 처리해 준다[1].

ATM 스위치는 대체로 반안 네트워크를 기반으로 구성되며 성능 변수들로는 셀프 라우팅, 이용률, 지연, 처리율 등이 있다. 이러한 네트워크 구조는 단일 경로를 가지기 때문에 두개이상의 셀이 스위치 소자 내에서 같은 출력포트로 나아갈 때 출력쪽에 블로킹이 일어날 수 있으며 이러한 블로킹으로 인해 셀의 처리율과 손실에 영향을 주게 된다. 해결방법으로는 첫째, 스위치 소자의 내부에 입·출력 버퍼 또는 공유 버퍼를 사용하는 것과 둘째, 상호 연결 네트워크에서 링크의 속도를 증가시키는 방법과 셋째, 블로킹이 발생할 수 있는 셀들을 미리 감지하여 feed-back이나 back-pressure시키는 방법이 있으며 넷째는 MIN을 병렬로 구성하여 입·출력간의 선택적 경로를 제공하는 방법이 있고, 다섯째는 정렬-반안 네트워크를 이용하는 방법 등이 있다[2][3]. 현재까지 블로킹 문제를 해결하기 위해서 정렬-반안 네트워크를 사용하는 방법을 고려하며 전단에서는 배치 이진 정렬 네트워크가 패킷을 목적지에 따라 먼저 분류하고 다음 단에서는 전달된 셀을 반안 네트워크로 라우팅 시켜서 셀을 재 정

렬해 주므로서 반안 네트워크를 근본적으로 언블로킹 상태로 만들게 된다. 또한 이것은 버퍼를 사용한 스위치인 경우에 비해 스위치 노드간의 지연이나 지연의 불규칙성을 방지할 수 있는데 이러한 방법을 이용한 네트워크로는 Huang's Starlite 스위치, Hui's SIBS 스위치 등이 있다. 그러나 배치 이진 정렬 네트워크의 단점으로 전체 네트워크의 전단에 배치 이진 정렬 네트워크를 추가함으로써  $\log_2 N(\log_2 N + 1)/2$  (N=입·출력수) 만큼 스테이지가 증가하고 스위치 소자도 N/2 만큼의 하드웨어적인 요소의 증가로 이어져 많은 스위치 소자와 링크에서 오류가 발생할 경우 정렬 네트워크의 기능을 손상시킬 수 있다. 즉, 전단의 정렬 네트워크의 오류로 인해 후단의 반안 네트워크의 언블로킹 상태를 파괴할 수도 있는 단점을 가지고 있다[4][5].

본 연구에서는 전단의 배치 이진 정렬 네트워크에서 서브 스위치 소자를 이용하여 오류 허용 경로를 추가하는 것으로 특정 스위치 소자나 링크에서 오류가 발생하더라도 여분의 경로로 라우팅이 가능한 구조를 제안하고자 한다. 기존의 네트워크 구조로는 병렬 정렬 네트워크(PSN:Parallel sorting network)와 오류 허용 정렬 네트워크 (FTSN:Fault-tolerant sorting network)가 있는데 이들 경우 오류 허용 경로는 제공할 수 없지만 하드웨어적인 복잡성과 네트워크 크기의 증가시 오류 허용 경로의 수가 부족한 문제가 있다. 본 연구에서는 배치 이진 정렬 네트워크의 하드웨어적인 복잡성을 줄이면서 오류에 대한 오류 허용 경로를 증가시키는 네트워크를 제안하고자 한다.

2. 오류 허용 정렬 네트워크  
2.1 병렬 정렬 네트워크(PSN)

PSN은 배치 이진 정렬 네트워크를 두 개를 사용한 것으로, 한 plane에서 오류가 발생할 경우 다른 plane으로 경로가 대체된다. 그림 1에서 PSN의 동작은 메시지를 보내기 전에 오류 없는 정렬 plane을 먼저 확인한 후 경로를 결정을 하는데 결정된 plane은 입력 정렬을 위해 사용된다. 셀은 출력포트에 도착하기까지  $(n \cdot (n+1))/2$  스테이지를 통해 전달되며 이로 인해 발생하는 여분 plane은 오류 허용 경로를 확보해 준다. 그러나 PSN의 단점으로는 두 개의 plane에서 한쪽만 셀 전송에 사용이 되므로 네트워크가 크기 변화에 따른 하드웨어적 복잡도가 증가한다는 점이다[6][7].

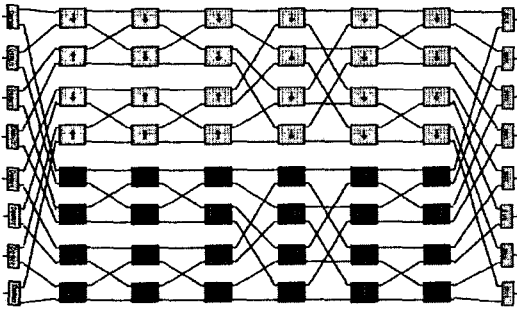


그림 1 Parallel sorting network (N=8)

2.2 오류 허용 정렬 네트워크(FTSN)

그림 2의 네트워크에서 두개의 정렬 네트워크는 상호 교차하면서 연결이 된다. PSN에서 셀은 한 번만 특정 plane으로 보내지는 반면에 FTSN에서는 셀이 전송되다가 오류가 발생하면 두 개의 planes에서 상호 오류 허용 경로를 제공하므로 다른 plane으로 전환이 되는데, 이 때 사용된 네 경로중 동일 plane의 두 개는 스테이지 i에서 다음 스테이지 i+1로 연결되고, 두 개는 다른 plane의 스테이지 i+1로 간다. 또한 FTSN의 스테이지는 0에서부터  $(n \cdot (n+1))/2 - 1$ 까지, 스위치 소자는 0에서부터  $N/2 - 1$ 를 사용한다. 만약 메

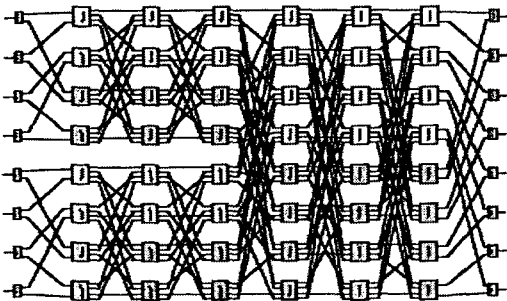


그림 2 Fault-tolerant sorting network (N=8)

시지가 plane 0에서 네트워크에 입력되어 메시지가 전송되는 동안 오류가 발생하면 메시지는 plane 1로 bypass하고, 다음 스테이지에서 셀은 plane 0으로 다시 전송된다. 그러나 링크와 여분의 plane에 의한 하드웨어의 증가로 역시 복잡도를 상승시킨다[8][9].

3. 제안된 구조와 성능평가

3.1 제안된 네트워크 구조

제안된 네트워크 구조는 그림 3과 같이  $2 \times 2$  스위치 소자를 사용한 기본 배치 이진 정렬 네트워크 구조에서 각 스테이지 사이에 서브 스위치를 추가한 형태이다. 추가된 서브 스위치는 다음 스테이지에 있는 스위치 소자에 연결하는 것에 사용이 된다. 이때 제안된 구조의 경로는  $3^{(n+1)/2}$  을 가지게 되는데 이 때 각 스테이지에서 추가된 스위치 소자는  $\sum_{k=0}^{n/2} \binom{n}{k} \cdot \frac{1}{2}$  이며, 전체 스테이지에서 사용된 스위치 소자는  $((n^2+n) \cdot \frac{N}{4}) + ((n^2+n) \cdot \frac{N}{8} - \frac{N}{4}) + \frac{N}{2}$  이다. 만약에 셀이 정상 스위치 소자에서 오류가 생기면 s1의 서브 스위치에 의해 여분의 경로로 전송이 된다. 그리고 이 셀은 서브 스위치의 라우팅에 의해 다음 스테이지의 스위치 소자로 전송이 된다. 그런데 s1의 서브 스위치에서 또 다시 오류가 생기면 s2의 서브 스위치로 이동이 되어 셀이 전송이 된다.

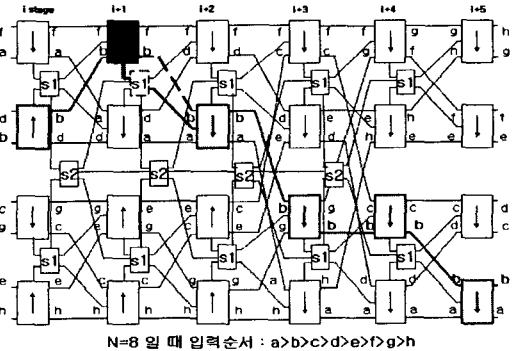


그림 3 오류시 여분의 경로로 라우팅 (N=8)

그림 3에서는 라우팅의 입력 단에 들어오는 셀 순서를 임의로 a-h까지 정하고 각 셀이 각 스테이지를 이동할 때마다 각 스위치 소자가 오류에 대한 정보를 검사하여 오류가 없을 경우에는 다음 스테이지의 정상적인 경로로 라우팅이 되지만 특정 스위치에서 오류가 발생했을 경우에는 같은 스테이지의 s1에 의해 여분의 경로로 셀이 라우팅이 되게 한다. 위에서 들어오는 입력 셀 중에서 d번째 입력셀은 첫 스테이지에서 정상 스위치에 의해 라우팅이 되었지만 둘째 스테

이지인  $i+1$ 에서는 내부  $2 \times 2$  스위치의 오류로 여분의 경로를 제공해 주는  $s1$ 로 bypass되어지고 다시  $s1$ 의 내부 라우팅에 의해 다음 스테이지인  $i+2$ 의 정상 스위치로 라우팅이 된다.

표 1 제안된 라우팅 알고리즘

```

Begin
   $S_{n-1} S_{n-2} \dots S_0$ 
  Initial variable
   $P_k$ : routing of arrows
   $[P_k]s^i$ : //s: sub-switch, i: stage(0 to n-1)
  SE: [ $p_{n-1} p_{n-2} \dots p_{n-1+i} p p_{n-1} \dots p_{i+1}$ ]  $s-1^{i+1}$ 

  If  $[P_k]s^i$  is no faulty SE then
    send cell to  $[P_k]s^{i+1}$ 
  Else
    send cell to  $[P_k]s+1^i$ 

  If  $[P_k]s^i$  is faulty SE then
    send cell to  $[P_k]s+1^i$ 
  Else
    send cell to  $[P_k]s^{i+1}$ 
End
    
```

표 1은 각 스테이지에서 셀이 전송되는 라우팅 알고리즘을 보여준다.  $[P_k]s^i$ 는 스위치의 위치를 나타내며  $[p_{n-1} p_{n-2} \dots p_{n-1+i} p p_{n-1} \dots p_{i+1}]s-1^{i+1}$ 에서  $p$ 는 상하 단의 라우팅을 결정하는 스위치의 신호로 사용이 된다. 먼저  $[P_k]s^i$ 에 오류가 없으면 셀은 정상적인 경로인  $[P_k]s^{i+1}$  전송이 되고, 아니면 서브 스위치로 이동한다. 그리고  $[P_k]s^i$ 가 오류이면 셀은 서브 스위치  $s1$ 인  $[P_k]s+1^i$ 으로 가고, 아니면 정상 경로로 라우팅 된다.

3.2 스위치 소자와 동작원리

제안된 구조에서는  $2 \times 2$ ,  $2 \times 3$ ,  $3 \times 3$ ,  $3 \times 2$  등 4가지 종류의 스위치 소자를 갖는다. 그림 4(a)는  $2 \times 2$  이진 정렬 스위치 구조로 중앙의 화살표에 의해서 결정되어지며 입력 A, B값이 들어 올 때 출력  $H=\max(A, B)$ 일 때 연결되고, 출력  $L=\min(A, B)$ 일 때 연결이 된다. 그림 4(b)에서는  $2 \times 3$  스위치 소자는 내부에서 두 개의 입력을 Demux에 의해  $2 \times 2$  스위치 소자로 연결이 되어진다. 그러나 FD(Fault-Detection)/OC(Output Controller)에서  $2 \times 2$  스위치 소자의 오류 상태를 검사하여 오류시에서는 아래쪽의 여분 경로로 출력되어져 추가된 서브 스위치로 연결이 되어진다. 그림 4(c)에서 보여주는  $3 \times 3$  스위치 소자는 네트워크의 대부분의 내부 공간에서 사용이 되는 것으로 네트워크에서 셀 라우팅 기능을 담당한다. 동작순서는 먼저 상하 단의 정상입력과 앞 스테이지에서 오류에 의해 입력된

셀은 LC (Label checker)에 의해 상·하단으로 분화되어진다. 분화된 셀은 Mux에 다시 하나로 합쳐져서 다시 Demux에 의해 분산이 된다. 만약 내부  $2 \times 2$  이진 정렬 스위치 소자가 오류가 발생했을 경우에는 여분 경로로 출력이 이루어져서 둘째 서브 스위치인  $s2$ 로 셀이 이동을 한다. 그림 4(d)에서 보여주는  $3 \times 2$  스위치소자 구조는  $s2$ 에서 들어오는 셀이 정상적으로 들어오는 셀 입력단과 만나서 LC에 의해 검사되어져 상·하 단의 Mux로 연결이 되어 뒤쪽의  $2 \times 2$  이진 정렬 스위치 소자에 연결이 된다.

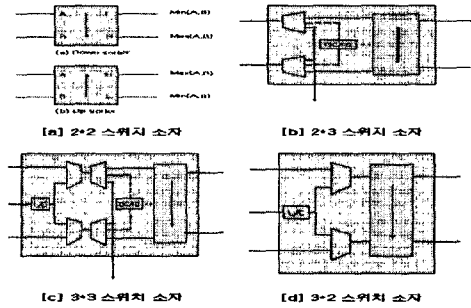


그림 4 4개의 정렬 스위치 소자

그림 5에서는 서브 스위치 소자에서 사용된 FD/OC와 L/C의 구조를 설명하고 있다. 먼저 그림 5(a) FD/OC 제어 순서도를 보면 FD에서  $2 \times 2$  스위치 소자에 오류상태를 파악하는 메시지를 전송하여 오류에 대한 정보가 yes이면 출력 OC에 목적지지를  $d=1$ 로 주어 Demux의 하단인  $d1$ 으로 출력이 나가고 No이면 Demux의 상단으로 정상적으로 셀이 전송되어  $2 \times 2$  스위치 소자를 통과한다. 그림 5(b) L/C 제어 순서도를 보면 먼저 앞 스테이지에서 오류상태의 셀을 받아서 P값의 정보에 의해서  $P=H$ 이면 상단 Mux로 아니면  $P=L$ 일 경우 하단 Mux로 셀이 전송이 된다. 이를 통해 앞 스테이지에서 전송된 셀은 스위치의 제어 흐름도를 따라서 목적지까지 전송이 된다.

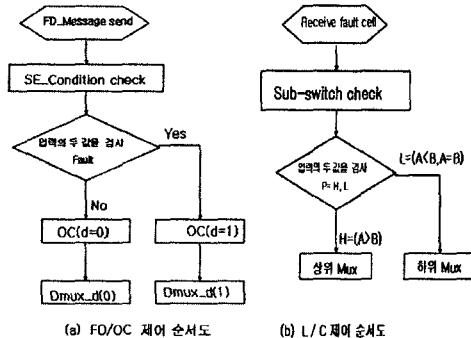


그림 5 LC와 FD/OC의 제어 흐름도

3.3 성능평가

정렬 네트워크에서 셀이 출발지에서 목적지까지 전송되는 과정에서 여분의 경로 수, 스위치 수, 그리고 링크 수를 통한 비용 분석을 하고자 한다. 표 2는 네트워크 크기를 N=8의 경우 각 네트워크 별로 사용되는 스위치 소자 수, 링크 수를 나타내고 있다. 표에서 제안된 구조가 기존의 다른 네트워크와 비교해서 스위치 소자 수와 사용된 링크 수가 적은 것을 알 수 있으며, 네트워크의 크기가 증가함에 따라 그 차이는 더욱 커짐을 알 수 있다.

표 2 네트워크의 비용 분석 (N=8)

네트워크 구조	Stage 수	SE의 수	LINK 수
BSN	6	24	56
PSN	6	64	128
FTSN	6	64	208
Proposed	6	38	115

그림 6은 BSN, PSN, FTSN 및 제안된 구조에서의 여분의 경로를 표시한다. BSN는 입·출력 사이에서 단일 경로를 주고, PSN는 입·출력 사이에서 두 경로를 준다. 그러나 각 plane에서 스테이지는 단일 경로를 준다. 그리고 FTSN에서는 plane의 각 스테이지에서 두 경로를 준다. 만약 스위치 소자에서 메시지가 오류를 만나면 다음 스테이지에서 메시지는 다음 plane으로 보내진다.

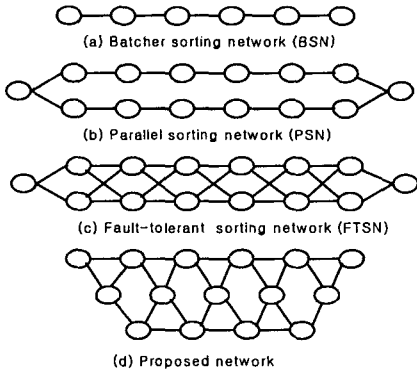


그림 6 여분 경로 그래프 (N=8)

본 연구에서 제안한 서브 스위치를 이용한 이진 정렬 네트워크의 여분 경로는  $3^{n(n+1)/2}$  으로 각 스테이지에서 세 곳의 경로를 주어진다. 이때  $n(n+1)/2$ 는 스테이지 수를 나타내는 식이다. 이로 인해 제안된 구조는 스테이지 수의 제곱으로 경로를 설정한다. 표 3은 네트워크 크기 변동에 따른 여분의 경로 수를 나타낸다. 여기서 n은 반얀 네트워크에서의 스테이지의 수를 나타낸다. 네트워크 크기가 증가함에 따라 제안된 구조는 점점 경로 수가 증가하는 것을 볼 수 있다.

표 3 여분의 경로의 수 (N=8)

종류	경로 수	n=3	n=4	n=5
BSN	1	1	1	1
PSN	$2^{k-2}$	2	2	2
FTSN	$2^{n(n+1)/2}$	64	1024	$3.2 \times 10^4$
Proposed	$3^{n(n+1)/2}$	729	59049	$1.5 \times 10^7$

4. 결론

기존의 제안된 PSN, FTSN의 경우 스위치 수와 링크 수등의 하드웨어적 비용의 증가로 복잡도가 높아지는 문제점과 네트워크 크기에 따른 오류에 대한 오류 허용 경로 수의 부족 현상이 나타난다.

본 연구에서 제안한 구조에서는 정렬 네트워크에서 각 스테이지마다 서브 스위치를 사용하여 여분의 경로 수를 증가시키는 방법을 제안하고 있다. 이것은 기존 네트워크보다 오류 허용을 위한 라우팅 경로가 증가한 반면 사용된 스위치 소자, 링크 수 등 하드웨어적인 복잡도는 감소시키고 있다.

5. 참고 문헌

- [1] JOHN WILEY & SONS, "Switching Theory", Achille Pattavina, Italy, 1998.
- [2] Ra'ed Y. Awded, H.T. Mouftah, "Survey of ATMswitch architectures", Elsevier Computer Networks 27, pp. 1567-1613, 1995.
- [3] C.-C. Lo, C.-Y. Chiou, "A Fault-tolerant architecture for ATM networks", Computer Communication 22, pp.1540-1548, 1999.
- [4] S. Kamolphiwong, A. E. Karbowski, H. Mehrpour, "Survey Flow control in ATM networks : a survey", ELSEVIER Science Computer Communications 21, pp. 951-968, March 1998.
- [5] Daniel Sobirk, Johan M Karlsson, "ATM Switch Structures - A Performance Comparison", 1994.
- [6] Neeraj K. Sharma, "Fault-Tolerant Sorting Network for ATM Switching", IEEE Communication, vol.7, pp.2075-2079, 1995.
- [7] Muhamad Anan, Mohsen Guizani, "A fault-tolerant ATM switching architecture" IEEE IPCCC, pp. 295-301, 2000.
- [8] Neeraj K. Sharma, Pierre U. Tagle, "Performance of Fault-Tolerant sorting network ATM switch", IEEE Communication vol.34, pp.91-107, 1998.
- [9] Jianli Sun, E. Cerny, J. Gecsei, "Fault-Tolerance in a Class of Sorting Networks", IEEE Transaction on computers, vol.43, No.7, July, 1994.