링망을 이용한 병렬 멀티캐스트 패킷스위치에서의 멀티캐스트 셀 스케줄링에 관한 연구

김 진 천**

The study on Multicast Cell Scheduling for Parallel Multicast packet switch with Ring Network

Jin-Chun Kim

요 약

오늘날의 통신서비스들은 음성이나 데이터 서비스와 같은 전통적인 서비스뿐만 아니라 비디오 서비스와 같은 대량의 데이터를 전송해야만 하는 멀티미디어 서비스를 포함한다. 이러한 요구를 수용하기 위해서 BISDN(Broadband Integated Services Digital Network)이 개발되었고 이의 기반 기술로 ATM이 채택되었다. 다양한 멀티미디어 응용들 중에서 VOD(Video On Demand), 화상회의등은 데이터를 동시에 여러 목적지로 보내는 멀티캐스트 특성이 있다. 그러므로 멀티캐스트 능력은 멀티미디어 통신에서 매우 중요하다. 본 논문에서는 링망을 이용한 병렬 멀티캐스트 패킷스위치(Parallel Multicast Packet Switch with Ring Network: PMRN)에서의 분리된 HOL을 이용한 멀티캐스트 셀 스케줄링 기법을 제안한다. 이기법은 입력버퍼의 앞단에 멀티캐스트 셀과 유니캐스트 셀을 위한 분리된 HOL을 두고 non-FIFO방식을 사용함으로서 입력버퍼 내에서 전송 가능한 멀티캐스트 셀과 유니캐스트 셀을 동시에 스케줄링할 수 있도록 하여 입력버퍼 내에서의 지연을 감소시키고 링망과 일대일 연결 네트워크의 이용율을 높이며 스위치의 처리율을 높일 수 있다.

ABSTRACT

A goal of a BISDN network is to provided integrated transport for a wide range of applications such as teleconferencing, Video On Demand etc. There require multipoint communications in addition to conventional

* 본 연구는 1998학년도 경성대학교 학술지원 연구비에 의하여 연구되었음.

** 경성대학교 전기전자 컴퓨터공학부

접수일자 : 2000. 4. 18.

point-to-point connections. Therefore multicast capabilities are very essential in multimedia communications. In this paper, we propose a new multicast cell scheduling method on the Parallel Multicast Packet Switch with Ring network: PMRN which are based on separated HOL. In this method, we place two different HOLs, one for unicast cells and the other for multicast cells. Then using non-FIFO scheduling, we can schedule both unicast cells and multicast cells which are available at the time in the input buffer. The simulation result shows that this method reduces the delay in the input buffer and increases the efficiency of both point-to-point network and ring network, and finally enhances the bandwidth of the overall packet switch.

I. 서 론

오늘날의 통신서비스는 음성이나 문자 위주의 데이터 서비스를 포함할 뿐만 아니라 그림이나 동영상과 같은 대량의 데이터를 고속으로 전송하여야 하는 멀티미디어 통신서비스가 요구되고 있다. 이러한 멀티미디어 통신에서의 요구사항으로는 고속의 데이터 전송 능력과 bit rate의 변화를 지원할 수 있는 능력 그리고 서로 다른 정보 형태의 동기화가 필요하다. 그리고 이러한 서로 다른 정보 형태들을 위해 다양한 서비스 품질을 제공할 수 있어야 한다.

또한, 오늘날의 통신 서비스중 그 중요도가 높이 부각되고 있는 주문형 비디오 서비스와 온라인 화상 회의 등을 위해서는 멀티캐스트 트래픽을 처리할 수 있는 능력이 꼭 필요하다. 이러한 복잡한 요구들과 새로운 통신의 개념들을 수용하기 위해서는 현재의 협대역 서비스를 위한 N-ISDN (Narrowband Integrated Services Digital Network)과 같은 음성을 주체로 고안된 네트워크는 한계가 있다. 그래서 검토되기 시작한 것이 광대역 서비스를 위한 B-ISDN (Broadband Integrated Services Digital Network)이다[1]. 1980년대 동안에 협대역 서비스(narrowband service)와 광대역 서비스(broadband service)를 함께 지원할 수 있는 능력을 가진 패킷 스위치(packet switch)에 대한 연구가 전세계적으로 행해졌다.

B-ISDN은 ATM(Asynchronous Transfer Mode)기술을 기반으로 하는 독특한 전송망으로 전개되었다. ATM기술은 오늘날의 모든 서비스뿐만 아니라, 미래의 협대역(narrowband)과 광대역(broadband)서비스를 모두 지원할 수 있는 유연성이 있다[2]. 본 논문에서는 멀티미디어 통신을 위한 다양한 요구 사항들 중에서 ATM 스위치 상에서의 멀티캐스트 트래픽의 처리 능력을 높이는데 주안점을 두고 연구를 하였다.

ATM은 Cell Switching 개념의 network으로 53byte의 고정된 크기의 셀을 사용한다[3]. ATM 스위치는 동일 출력단에 대한 입력셀들의 경쟁으로 인해 발생하는 스위치내의 대기를 위한 버퍼의 위치에 따라 입력버퍼 스위치, 출력버퍼 스위치, 그리고 공유버퍼 스위치로 나눌 수 있다. 입력버퍼 스위치일 경우 H/W 구현 복잡도가 낮아 스위치의 대규모화와 고속화에 알맞지만 FIFO방식으로 동작할 경우 출력단 충돌로 인해 전송되지 못한 입력단의 선두의 셀에 의해 입력버퍼내의 다른 셀의 전송을 방해하는 HOL(Head Of Line)블럭킹으로 인해 0.586의 낮은 처리율을 갖는다[4]. 공유 버퍼 스위치는 메모리 접근대역폭에 의해 최대 처리율에 제한을 갖는다. 출력버퍼 스위치는 처리율 면에서 가장 좋은 성능을 가지지만 내부속도를 올려야 하므로 구현 복잡도가 높아지는 단점이 있다.

입력버퍼 ATM 스위치에서 FIFO방식으로 인한 처리율의 한계를 극복하기 위해 입력버퍼의 선두 이외의 셀을 전송하는 non-FIFO방식이 연구되었다. 입력버퍼 ATM 스위치에서의 셀 스케줄링은 입력버퍼의 셀들중 출력단으로 동시에 전송 가능한 최대의 셀들의 집합을 찾는 문제이다. 그리고 입력버퍼로 들어오는 셀들은 두 대의 단말간의 일대일 전송을 기반으로 하는 유니캐스트(unicast)셀과 하나의 단말에서 여러개의 단말로 동시에 전송되어지는 멀티캐스트(multicast)셀이 있다.

기존의 유니캐스트 셀 스케줄링 알고리즘에서의 처리율 최대화 기법으로는 각 입력버퍼 앞단에 윈도우를 설정하고 윈도우 내에서 서로 다른 출력단을 갖는 셀들의 집합을 찾는 윈도우 기법과 윈도우 내에 있는 셀들과 출력단을 순차적으로 정합하는 윈도우 기법과는 달리 정합과정을 병렬적으로 수행함으

로써 필요한 정합횟수를 줄인 반복 병렬 정합기법[5], 그리고 입력버퍼내의 셀들중 임의의 시점에서 전송집합을 실제 전송시점 보다 빠른 시점들에서 파이프라인 방식으로 선택함으로써 전송집합 선택과정을 고속화한 시간예약기법이 있다.[6]

ATM 스위치에 대한 연구는 유니캐스트 스위치뿐만 아니라 온라인 화상회의나 VOD(Video On Demand)로써 최근에 화제가 되고 있는 영상 분배 서비스 등의 멀티캐스트 특성을 갖는 서비스를 효율적으로 제공해줄 수 있는 멀티캐스트 스위치에 대해서도 이루어지고 있다. ATM 통신의 기본적인 통신 형태는 2개의 단말 사이의 일대일(point-to-point) 통신이다. 1개의 ATM 단말로부터 송출한 데이터를 복수의 단말이 수신할 수 있도록 하기 위해서는 ATM 스위치에서 셀 복제(copy) 기능을 갖게 할 필요가 있다.

이러한 멀티캐스트 트래픽을 위한 스위치 구조에 관한 기존의 연구는 많이 이루어져 왔지만 셀 스케줄링 알고리즘에 대한 연구는 거의 이루어져있지 않다.

멀티캐스트 셀 스케줄링 알고리즘은 하나의 멀티캐스트 셀이 여러 셀 타임에 걸쳐 출력단으로 복사, 전송될 수 있는 목적지 분할(fanout splitting) 방식과 하나의 멀티캐스트 셀의 복사, 전송은 한 셀 타임만에 이루어져야 하는 목적지 비분할(fanout non-splitting)방식으로 나눌 수 있다[7].

본 논문에서는 링망을 이용한 병렬 멀티캐스트 패킷 스위치(Parallel Multicast Packet Switch with Ring Network: PMRN)[8]에서의 분리된 HOL을 사용한 멀티캐스트 셀 스케줄링 기법을 제안한다. 이 기법은 입력 버퍼의 앞단에 멀티캐스트 셀과 유니캐스트 셀을 위한 분리된 HOL을 두고 non-FIFO 방식을 사용함으로써 입력버퍼 내에서 전송 가능한 멀티캐스트 셀과 유니캐스트 셀을 동시에 스케줄할 수 있도록 하여 입력 버퍼에서의 지연을 감소시키고 링망과 일대일 연결 네트워의 이용률을 높이며 스위치의 처리율을 높일 수 있다.

본 논문의 구성은 다음 같다. Ⅱ장에서는 관련연구로서 지금까지 연구된 스위치에서의 셀 스케줄링 알고리즘에 대해 살펴보고, Ⅲ장에서는 본 논문에서 제안하는 수정된 PMRN스위치에서의 셀 스케줄링 기법에 대해 기술하며, Ⅳ장에서는 제안한 기법에 대하여 시뮬레이션을 통해 성능을 검증한다. 그리고 마지막으로 Ⅴ장에서 결론을 내리고 앞으로의 연구 방향을 제시한다.

II. 관련연구

이번 장에서는 데이터를 고속으로 처리하기 위해 개발된 ATM 스위치의 구조와 ATM 스위치에서의 처리율 향상을 위한 기법인 셀 스케줄링 알고리즘에 대해 살펴본다. ATM에 대한 연구는 스위치 자체에 대한 연구뿐만 아니라 ATM 스위치로 들어오는 다양한 트래픽을 효과적으로 처리하여 처리율을 높이기 위해 스위치 수준에서의 다양한 셀 스케줄링 알고리즘에 대한 연구도 진행되어 왔다.

2.1 ATM 스위치의 구조

ATM 스위치에서 채택된 구조는 멀티스테이지(multistage)로 구성된 간단한 SE(Switching Element)들과 셀의 헤더에 기록된 정보만으로 스위치 내에서 출력단을 찾아갈 수 있는 자가 라우팅(self routing)을 위한 IN(Interconnection Network)으로 구성된다. 그리고 IN은 타임 슬롯(time slot)을 사용하여 동기화되어 있다. 이러한 MIN(Multistage Interconnection Network)은 기본적으로 충돌(blocking)형과 비 충돌(nonblocking)형으로 분류된다. 충돌형은 서로 다른 입출력 경로가 스위치의 내부 경로(interstage path)를 공유하게 되어 셀 손실이 발생하게 되고, 이를 피하기 위해서 새로운 기술이 필요하게 된다. 반면에 비 충돌형은 서로 다른 입출력 경로가 분리되어 있다. 그래서 SE는 내부 버퍼가 필요하지 않다. 하지만 충돌 형보다 많은 스테이지가 필요하다. 본 논문에서는 비 충돌 형태의 스위치를 대상으로 연구를 하였다. 비 충돌 형태의 ATM 스위치는 동일 출력단에 대한 입력 셀들의 경쟁으로 인해 발생하는 스위치내의 셀들의 대기를 위한 버퍼의 위치에 따라 입력버퍼 스위치, 출력버퍼 스위치, 그리고 공유버퍼 스위치 그리고 앞의 구조들의 혼합된 형태의 스위치로 나눌 수 있다.

2.1.1 입력버퍼 스위치

입력버퍼 스위치는 스위치 앞단에 출력단 경쟁

에서 진 셀들의 대기를 위한 버퍼가 있다. 입력버퍼 스위치의 경우 H/W 구현복잡도가 낮아 스위치의 대규모화와 고속화에 알맞지만 FIFO방식을 사용함으로써 발생하는 HOL(Head Of Line)블럭킹으로 인해 0.586의 낮은 처리율을 갖는다.

2.1.2 공유버퍼 스위치

공유버퍼 스위치는 공유 메모리(shared memory)를 사용하여 입력된 셀을 버퍼링한다. 공유버퍼 스위치는 공유 메모리를 사용함으로써 집중된 컨트롤을 할 수 있어 편리하지만 메모리 접근 속도에 한계가 있어서 최대 처리율에 제한을 갖는다.

2.1.3 출력버퍼 스위치

출력버퍼 스위치는 스위치의 뒤에 버퍼를 두고 스위치 내부의 속도를 높여 동시에 같은 출력단으로 가려는 셀을 출력버퍼에 넣는다. 이러한 N×N 출력버퍼 스위치는 스위치내의 속도증가 k ($1 \le k \le N$)로 특성화될 수 있는데, 여기에서 속도 증가 k 는 같은 타임 슬롯 동안에 각 출력버퍼에 받아들여질 수 있는 셀의 수를 나타낸다. 출력버퍼 스위치는 HOL블록킹이 없어 처리율이 높아지는 반면에 내부 속도 증가로 인해 구현 복잡도가 높아지는 단점이 있다.[4]

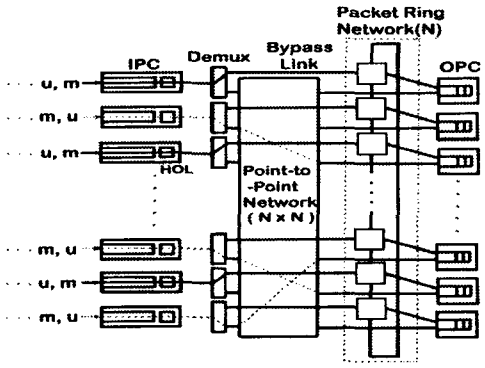
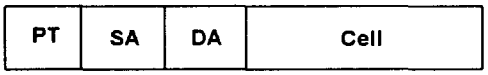


그림 2.1. 링망을 이용한 병렬 멀티캐스트 패킷스위치

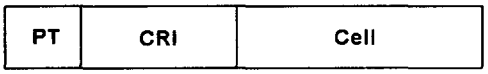
2.1.4 링망을 이용한 병렬 멀티캐스트 패킷 스위치

입력버퍼와 출력버퍼의 혼합된 형태의 스위치로 그림 2.1에서와 같은 링망을 이용한 병렬 멀티캐스트 패킷 스위치(Parallel Multicast packet switch with Ring Network :PMRN)[8]가 있다. PMRN 스위치는 N개의 병렬로 된 노드를 가진 N × N 일대일 연결 스위치망(PSN)과 패킷 링망(PRN)으로 구성된다. 또 다른 구성 요소에는 입력 단자 제어기(IPC), demux, 우회 경로(Bypass link), 그리고 출력 단자 제어기(OPC)가 있다.



PT : Packet Type
SA : Source Address
DA : Destination Address

(a) Tag for unicast packet



PT : Packet Type
CRI : connection Request Information

(b) Tag for mlticast packet

그림 2.2. 셀 태그 구조

IPC는 입력버퍼에 들어온 패킷들을 유니캐스트 셀과 멀티캐스트 셀로 구분하고 그림 2.2에 나타난 바와 같이 유니캐스트 셀과 멀티캐스트 셀에 대한 알맞은 태그들을 가지고서 그 셀을 수정한다. 유니캐스트 셀에 대한 출발지와 목적지 주소는 일대일 연결망에서의 전송을 위해 추가된다. 그리고 멀티캐스트 셀에 대해서는 여러 개의 목적지 주소를 지정하지 않고 특정 출력단자에의 연결은 CRI(Connection Request Information)필드의 해당 bit의 위치에 1로 나타낸다. 그리고 이 태그를 이용하여 demux에서는 입력버퍼의 선두의 셀들중 유니캐스트 셀은 일대일 연결망으로 보내고 멀티캐스트 셀은 우회 링크를 통해 링망으로 전송한다. 그리고 일대일 연결망과 링망은 서로 독립적으로 동작한다. 이렇게 멀티캐스트 셀과 유니캐스트 셀을 분리함으로써 멀티캐스트 셀과 유니캐스트 셀의 혼합된 트래픽을 효율적으로 다룰 수 있다.

PMRN 스위치에서는 병렬로 동작하는 두 개의 서로 다른 네트워크의 동작이 같은 타임 슬롯을 사용하여 동기화된다. 즉 τ를 타임 슬롯에 대한 클럭 사이클을 나타내는 것이라 할 때, PMRN 스위치는 τ초마다 셀을 수신하고 출력하며 PSN과 PRN은 이 시간 간격으로 동작한다.

PMRN 스위치에서의 셀 전송방법은 입력버퍼의 선두의 셀만이 고려된다. PMRN 스위치에서의 셀의 전송은 probe, acknowlegment, data의 세 단계에 의해서 스위치를 통해 패킷을 전송한다. probe단계에서 비어있지 않은 버퍼를 가진 각각의 IPC(Input Port Controller)는 셀의 헤더내의 출력단 주소를 사용, 유니캐스트 셀에 대한 출력 주소를 지정하여 request 셀을 나타내거나 혹은 demux와 우회 경로를 통해서 멀티캐스트 셀을 링망에 보낸다.
유니캐스트 셀의 경우에는 request가 ack. 단계에서 일대일 연결망에 의해 처리된다. 그리고 나서 ack. 셀을 받은 모든 IPC들은 data단계에서 그들의 HOL에 있는 유니캐스트 셀을 일대일 연결망을 통해 전송한다. ack. 를 받아들이는데 실패한 입력 단자들은 세 단계의 주기가 반복되는 다음 슬롯에서의 재시도를 위해 버퍼속에 셀을 보유한다. 반면에 멀티캐스트 셀을 위한 probe단계에서 링망에 보내진 멀티캐스트 셀은 링망을 통해 그들의 목적된 출력 단자에 전송되어진다. 각 멀티캐스트 셀의 헤더는 목적지 출력단자들에 대한 위치 bit를 포함하고 멀티캐스트 셀을 받은 링망내의 각 스위칭 노드는 버퍼들 속에 그것들을 로드시켜 놓은 후 스위칭 노드들을 통해서 동시에 이동시킨다. 모든 멀티캐스트 셀은 한번의 순환동안에 그것의 목적 노드에 복사될 수 있다. 결국 셀은 출력버퍼를 통해 목적지에 전송될 수 있다. 다음 사이클이 시작될 때 PRN으로의 멀티캐스트 셀의 전송이 재개된다.

그림 2.3은 이러한 셀의 전송 과정을 보여준다.

입력 버퍼의 선두의 셀이 멀티캐스트 셀이면 우회 경로를 통해 링망으로 보내어진다. 그리고, 유니캐스트 셀이면 다른 입력 버퍼의 유니캐스트 셀들과 출력단 경쟁이 없다면 바로 일대일 연결망으로 보내어지고, 출력단 경쟁으로 충돌이 발생한 버퍼들은 그 중에서 전송될 셀을 random하게 선택한다. 그리고, 출력단 경쟁에 진 셀들은 입력 버퍼의 선두에 남게 된다.

링망과 일대일 연결망은 병렬적으로 동작하므로 두 가지 망을 동시에 사용할 수 있음에도 그림 2.3에서 보는 것과 같이 입력 버퍼의 HOL의 셀 만이 스케쥴 되므로 선두의 셀이 유니캐스트 셀이면 입력버퍼의 뒤에 있는 멀티캐스트 셀은 현재의 타임 슬롯 동안 전송될 수 있음에도 선두의 유니캐스트 셀의 방해로 전송되지 못한다. 또한 FIFO방식을 사용함으로써 유니캐스트 셀들에 의해 발생하는 HOL블럭킹으로 처리율의 제한을 가져온다.

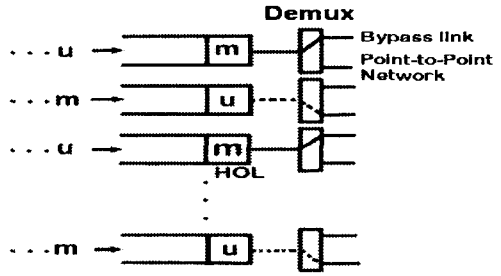


그림 2.3. PMRN 스위치에서의 셀의 전송과정

2.2 셀 스케쥴링 알고리즘

입력 버퍼 ATM 스위치에서의 셀 스케쥴링 알고리즘은 입력버퍼의 셀들중 출력단으로 동시에 전송이 가능한 최대의 셀들의 집합을 찾는 문제이다. 이는 출력단 경쟁으로 인해 발생하는 입력버퍼에서의 전송 지연을 줄이고 스위치 전체의 처리율을 높이고자 하는 것이다. 이러한 처리율 향상 기법은 유니캐스트뿐만 아니라 멀티캐스트 트래픽에 대해서도 연구가 진행중이다.
스위치로 들어오는 트래픽은 기존의 데이터 통신을 위한 유니캐스트 트래픽과 최근의 VOD, 화상회의등 데이터의 고속 전송과 다중 전송으로 특성지어지는 멀티미디어 통신을 위한 멀티캐스트 트래픽이 있다.

2.2.1 유니캐스트 셀 스케쥴링 알고리즘

입력버퍼 스위치에서의 스케쥴링 알고리즘은 크게 스케쥴시 버퍼의 HOL만을 고려하는 FIFO방식과

엄격한 FIFO방식을 완화하여 버퍼내의 다른 위치의 셀도 스케줄될 수 있는 non-FIFO방식이 있다. FIFO 방식은 출력단 충돌로 인해 전송되지 못한 입력단의 선두의 셀에 의해 입력버퍼내의 다른 셀들이 전송을 방해받아 발생하는 HOL블럭킹으로 그 처리율이 제한된다. 반면에 non-FIFO방식은 입력버퍼의 선두의 셀뿐만 아니라 버퍼내의 다른 셀들이 출력단 경쟁에 참여할 수 있어 HOL블록킹을 경감시켜 처리율을 높일 수 있다. 그러므로 non-FIFO방식을 사용하고 좋은 스케줄링 알고리즘을 사용함으로써 스위치 전체의 처리율을 높이고자 하는 연구가 많이 행해졌다.

2.2.1.1 FIFO 방식

FIFO방식에서는 주로 요청, 승인, 전송의 3단계를 가지는 삼 단계 (three phases)알고리즘을 사용한다. 입력포트 콘트롤러는 입력버퍼 선두의 셀들의 출력단 요청들 중에서 현재 타임 슬롯 동안 전송 가능한, 즉 충돌이 발생하지 않은 입력포트와 충돌이 발생한 입력포트들 중에서 하나를 선택하여 승인 신호를 보내고, 승인 신호를 받은 입력단은 그 타임 슬롯 동안 셀을 전송하고 출력단 경쟁에서 진 입력단의 셀들은 그림 2.4와 같이 입력버퍼에 남게 된다. 여기서 출력단 경쟁으로 충돌이 발생한 셀들 중에서 전송할 셀을 선택하는 방법으로 무작위 선택방법(Random selection policy), 고정 우선 순위 선택방법(Fixed priority selection policy), 그리고 가장 긴 큐 우선 순위 선택방법(Longest queue selection policy)이 있다[5].

무작위 선택 방법은 출력단 경쟁으로 충돌이 발생한 버퍼들 중에서 전송될 셀을 무작위로 선택하여 전송한다. 고정 우선 순위 선택 방법은 N개의 입력 버퍼가 고정된 우선 순위를 가지고 콘트롤러는 그중 가장 높은 우선 순위를 가지는 버퍼의 셀을 먼저 전송한다. 가장 긴 큐 우선 순위 선택 방법은 가장 긴 입력버퍼의 셀을 먼저 전송한다. 위 방법들 중에서 가장 긴 큐 우선 순위 선택 방법이 다른 방법들에 비해 셀의 대기 시간이 적다. 하지만 엄격한 FIFO방식에서는 버퍼의 HOL의 패킷만을 고려하므로 위의 어떤 방법을 쓰더라도 HOL블록킹으로 인해 처리율의 제한을 갖는다.

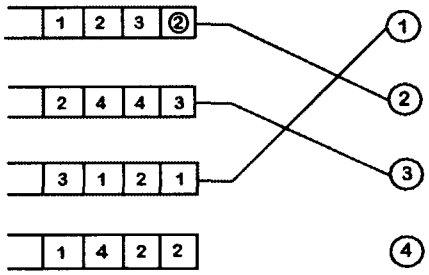


그림 2.4. FIFO 방식 스케줄링

2.2.1.2 non-FIFO방식

non-FIFO방식은 엄격한 FIFO방식을 완화하여 입력버퍼의 선두 이외의 셀을 선택함으로써 HOL 블록킹을 줄여 처리율을 높이고자 하는 방식이다. non-FIFO방식은 크게 현재 타임 슬롯 동안에 선택된 셀을 전송하는 비 예약 방식과 입력버퍼내의 셀들의 전송시간을 실제 전송시간에 앞서 예약하는 전송시간 예약 방식으로 나눌 수 있다.

비 예약 방식의 대표적인 예는 윈도우 기법[4]이 있다. 윈도우 기법은 그림 2.5에서와 같이 버퍼의 앞단의 w개의 셀들의 집합중 서로 다른 출력단을 갖는 셀들을 선택함으로써 스위치의 처리율을 높이고자 하는 것이다.

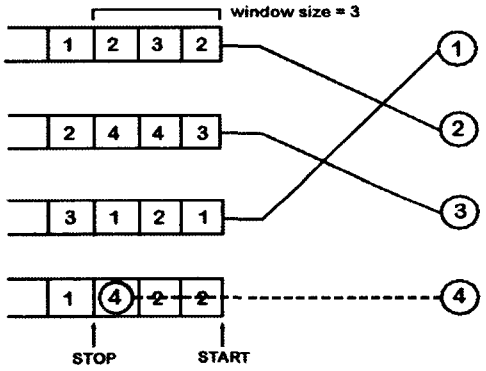


그림 2.5. Window 방식 스케줄링

여기서 'w'를 윈도우 크기라 하고 w번의 셀 선택과정을 가지게 된다. 그리고 k ($1 \le k \le w$) 번

째 셀들중 출력단 충돌이 없는 셀들을 선택한다. 이 방식은 16 × 16 스위치에서 w가 8인 경우 처리율이 최대 0.88까지 향상되지만, 셀 선택과정이 윈도우 크기만큼의 순차적인 반복과정으로 구성되어 윈도우가 커지면 처리 시간이 길어진다.

그림 2.6은 전송시간 예약 기법[6]에서 예약 과정을 보인 것이다. 경쟁 해소 모듈은 입력버퍼(IB)를 위한 예약 테이블(RT)을 가진다. RT에 할당된 i값은 1에서 N까지의 값으로 셀 타임 슬롯으로 나타난다. RTi는 다시 N개의 예약 상태를 나타내는 엔트리들로 구성되어지고 각 엔트리들은 출력포트와 관련되어진다. RTi(j)($1 \le j \le N$)의 값은 t가 현재 셀 슬롯일 때 t + i 시간에 출력포트 j의 예약상태를 나타낸다. IBi내의 셀은 RTi내의 출력포트 예약상태에 대응해서 매칭 되고 만약 매칭 되는 셀이 있다면 RTi(j)는 n에서 r로 바뀌어 예약되었음을 나타낸다. 그리고 매칭된 셀은 i 셀 슬롯 후에 스위치로 전송된다.

시간 예약 기법에서는 전송 시점의 예약이 각 입력단에서 독립적으로 수행된다는 장점이 있으나, 입력단마다 전송 대기시간이 다른 불공정성 문제와 낮은 부하에서도 최소 N/2의 평균 전송 대기시간이 걸리는 단점이 있다.

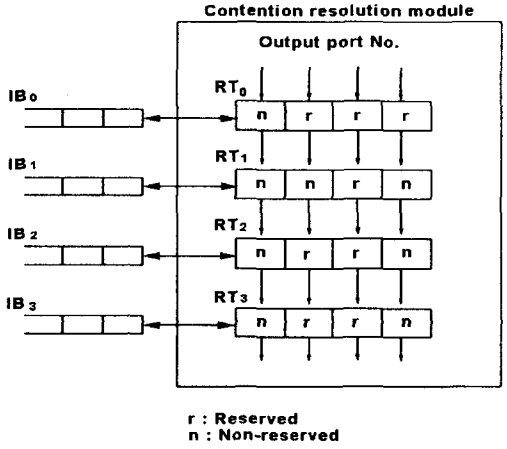


그림 2.6. 전송 시간 예약 방식 스케쥴링

2.2.2 멀티캐스트 셀 스케쥴링 알고리즘

멀티캐스트 셀은 하나의 셀이 동시에 여러 단말로 복사 전송되어야 한다. 이러한 멀티캐스트 셀을 전송하는 방식은 목적지 분할(fanout splitting)방식과 목적지 비분할(non-fanout splitting)방식으로 나눌 수 있다. 목적지 분할 방식은 하나의 멀티캐스트 셀이 여러 셀 타임을 걸쳐 전송되어질 수 있고, 목적지 비분할 방식은 하나의 멀티캐스트 셀은 한 셀 타임내에 복사, 전송이 이루어져야 한다. 이러한 멀티캐스트 트래픽을 위한 셀 스케쥴링 알고리즘에 대한 연구는 거의 이루어져 있지 않다. 특히 입력버퍼 ATM스위치에서는 주로 목적지 분할 방식의 스케쥴링 알고리즘이 연구 되어 왔다.

목적지 분할 방식의 멀티캐스트 셀 스케쥴링 알고리즘으로 멀티캐스트 라운드로빈(Multicast Round-Robin : mRRM)알고리즘[7]이 있다. 이 방식은 FIFO방식을 사용한다. FIFO방식의 스케쥴링에서 입력버퍼 선두의 멀티캐스트 셀들중 목적하는 모든 출력단으로 전송되지 못하고 남는 셀을 '나머지 셀(residue cell)' 이라 한다. McKeown[7]은 '나머지 셀'의 개수를 최소화함으로써 성능을 높일 수 있음을 보였다.

멀티캐스트 라운드로빈 알고리즘은 입력버퍼 선두의 셀만을 전송하는 FIFO방식을 사용한다. 먼저 각 입력버퍼 선두의 셀이 목적하는 출력단에 요구신호를 보내고, 각 출력단은 요구신호를 보낸 입력단 중 하나를 선택해 승인 신호를 보낸다. 이때 모든 출력단이 공유하는 라운드 로빈 포인터에서 가장 가까이 있는 입력단을 선택하게 된다. 라운드로빈 포인터는 스케쥴이 끝난 첫 번째 입력단 다음을 가리키도록 변경되고, 각 입력단은 승인신호를 받은 출력단으로 멀티캐스트 셀을 복사, 전송한다. mRRM에서는 모든 출력단이 하나의 라운드로빈 포인터를 공유하여 스케쥴링함으로써 '나머지 셀'의 개수를 줄이는 효과를 얻어 처리율을 향상시킨다. 그러나, 기본적으로 선두의 셀만을 전송하는 FIFO방식이므로, 처리율이 제약된다는 단점을 가진다.

mRRM방식에서 2 × N 스위치에서의 스케쥴 과정을 그림 2.7에 나타내었다. 화살표는 라운드로빈 포인터가 이동하는 방향, 즉 화살표가 가리키

는 방향으로 최소의 residue가 남도록 스케쥴 된다.

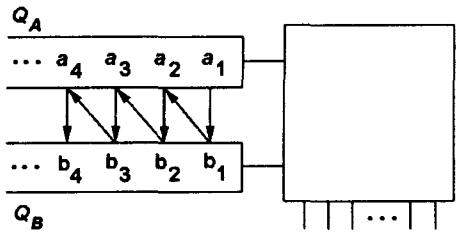


그림 2.7. 멀티캐스트 라운드 로빈 알고리즘

또 다른 목적지 분할 방식의 멀티캐스트 셀 스테쥴링 알고리즘으로 WBA (Weight Based Algorithm)이 있다. WBA방식[7]은 모든 셀타임의 시작에서 각 입력은 그것의 HOL에 있는 새로운 셀이나 나머지 셀의 가중치(*weight*)를 계산한다. 가중치는 HOL에 오래 머물러 있는 셀에 높게 주고 팬아웃이 큰 셀은 낮게 준다. 가중치 계산 후 각 입력은 가중치와 함께 가고자 하는 출력단에 요구신호를 보내고, 각 출력단은 가중치가 높은 입력을 승인한다. 이 방법도 HOL의 셀만을 전송하는 FIFO방식을 사용하므로 처리율에 제한이 있다.

Ⅲ. 수정된 링망을 이용한 병렬 멀티캐스트 스위치에서의 셀 스케쥴링

이 장에서는 본 논문에서 제안하는 수정된 PMRN 스위치에서의 분리된 HOL을 사용한 셀 스케쥴링에 대해 설명한다.

관련연구에서 언급한바와 같이 기존의 PMRN 스위치에서는 입력버퍼의 HOL의 셀만이 스케쥴링에 참여하는 FIFO방식을 사용함으로써 처리율에 제한을 가져온다. 특히 입력버퍼의 선두에 유니캐스트 셀이 들어왔을 때 현재 타임 슬롯 동안에 멀티캐스트 셀이 있다면 링망을 통하여 보낼 수 있는데도 불구하고 입력버퍼의 선두의 셀만을 고려하므로 보낼 수 없게 된다. 그러므로 non-FIFO방식의 스케쥴링 알고리즘을 사용하고 입력 버퍼내에서 멀티캐스트 셀과 유니캐스트 셀을 동시에 스케쥴할 수 있다면 유니캐스트 셀로 인한 HOL블럭킹을 감소시키고 동시에 링망과 일대일 연결망을 더욱 효율적으로 사용할 수 있게 되어 처리율을 향상시킬 수 있을 것이다.

3.1 수정된 링망을 이용한 병렬 멀티캐스트 스위치의 구조

그림 3.1은 본 논문에서 제안하는 수정된 PMRN 스위치의 구조를 보인 것이다. 기존의 PMRN 스위치에서는 입력 버퍼 뒤에 DEMUX를 두어 입력버퍼의 셀들을 멀티캐스트 셀은 우회 경로를 통하여 링망으로 그리고 유니캐스트 셀들은 일대일 연결망으로 보낸다. 그림에서와 같이 수정된 PMRN 스위치에서는 입력 버퍼내의 셀들중 현재 타임 슬롯동안 전송할 수 있는 유니캐스트 셀과 멀티캐스트 셀을 모두 선택할 수 있도록 demux를 제거하고 입력 버퍼에 분리된 HOL을 두었다. 선택된 셀들은 각각 분리된 HOL에 넣어지고 모든 입력단에서 선택이 완료되면 스위치를 통하여 전송된다.

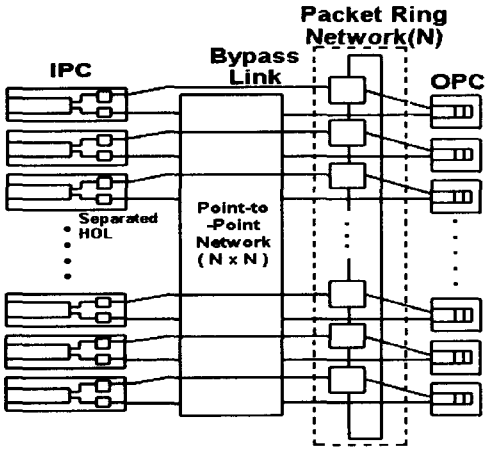


그림 3.1. 수정된 PMRN스위치의 구조

3.2 분리된 HOL을 사용한 셀 스케쥴링

수정된 PMRN스위치에서의 셀 스케쥴링은 하나의 입력 버퍼에서 전송 가능한 멀티캐스트 셀과 유니캐스트 셀을 모두 선택할 수 있도록 non-FIFO 방식을 사용한다.

분리된 HOL을 사용한 기법은 입력 버퍼의 앞단에

크기 w인 윈도우를 설정하고, 그 윈도우 내의 w 개의 셀들의 집합중 현재 타임 슬롯 동안 전송 가능한 멀티캐스트 셀과 유니캐스트 셀을 순차적으로 검색하여 모두 찾아낸다. 선택된 셀들은 각각 멀티캐스트 셀과 유니캐스트 셀을 위한 분리된 HOL에 넣어지고 모든 입력단에서 선택 과정이 완료되면 선택된 멀티캐스트셀과 유니캐스트 셀은 각각 링망과 일대일 연결망을 통하여 전송된다

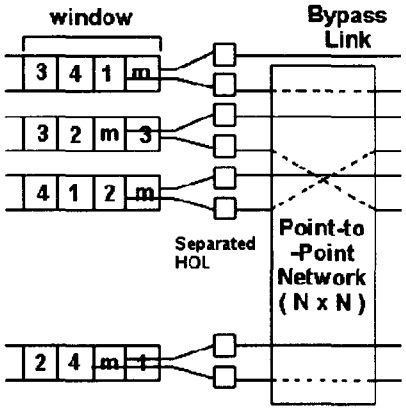


그림 3.2. 분리된 HOL을 사용한 스케쥴링

그림 3.2는 분리된 HOL을 사용한 스케쥴 과정을 보여준다. 그림에서와 같이 입력 버퍼의 앞단에 설정된 윈도우 크기내의 셀들중 입력버퍼에 가장 먼저 들어온 멀티캐스트 셀과 서로 다른 출력단을 갖는 유니캐스트 셀들을 선택한다. 그림 3.2에서 각각의 입력버퍼에서 현재의 타임 슬롯동안에 전송 가능한 멀티캐스트 셀과 유니캐스트 셀이 모두 스케쥴됨을 보인다. 첫 번째 버퍼에서는 입력버퍼의 선두의 멀티캐스트 셀과 1번 출력단으로 가는 유니캐스트셀이 선택되었다. 제일 마지막 버퍼에서는 입력버퍼 선두의 유니캐스트 셀이 출력단 경쟁으로 블록킹이 되었을 때 입력 버퍼의 윈도우 크기 내에서 세 번째 셀인 4번 출력단으로 가는 유니캐스트 셀이 선택되어 지는 것을 보여 준다.

이렇게 분리된 HOL을 사용함으로써 입력버퍼내의 멀티캐스트 셀과 유니캐스트 셀을 동시에 스케쥴함으로써 유니캐스트와 멀티캐스트의 혼합된 트래픽을 기존의 PMRN 스위치에 비해 효율적으로 처리하여 처리율을 높이고 병렬적으로 동작하는 링망과 일대일 연결망의 이용율을 높일 수 있다. 또한 입력 버퍼내의 셀들을 순차적으로 선택함으로써 동일 출력단으로 향하는 동일 입력단의 셀들은 입력단에 도착하는 순으로 출력단으로 전송할 수 있어 셀의 순서를 유지할 수 있다. 그리고 윈도우 크기가 커질수록 하드웨어 복잡도가 증가하고 전송 가능한 셀을 검색하는데 소요되는 시간이 늘어나지만 다음장의 시뮬레이션에서 적은 크기의 윈도우로 높은 처리율을 얻을 수 있음을 보인다.

Ⅳ. 성능 평가

본 장에서는 수정된 PMRN 스위치로 들어오는 트래픽이 랜덤 트래픽(random traffic)일 때의 경우를 시뮬레이션을 통해 조사하여 기존의 PMRN 스위치에서의 결과와 비교하여 제안한 알고리즘의 성능을 평가한다.

4.1 시뮬레이션 환경

시뮬레이션에서 윈도우 크기는 1에서 8까지, 스위치 크기는 16 × 16, fanout은 2와 5일 때를 가정한다. 그리고 시뮬레이터는 GNU C 2.8.1언어를 사용하여 작성하였고, 시뮬레이션은 Solaris 2.5.1을 운영체제로 사용하는 SUN사의 Enterprise 3000에서 수행하였다. 또한 스위치는 한 타임 슬롯을 단위시간으로 하였고, 입력버퍼는 유한버퍼를 가정하였으며 출력버퍼는 무한버퍼를 가정하였다. 결과는 전체 50000타임 슬롯을 조사한 것으로 처음 10000타임 슬롯은 스위치의 초기 상태의 영향을 배제하기 위해 결과에서 제외하였다. 성능평가의 척도로써 포화 상태에서의 처리율과 타임 슬롯 단위의 평균 지연을 조사하였다. 스위치로 들어오는 셀들의 트래픽 패턴은 랜덤 트래픽을 가정하였다. 랜덤 트래픽은 각 입력단으로 들어오는 셀은 도착율이 λ인 포아송 분포(Poisson distribution)를 따르며, 입력셀이 요구하는 출력단은 동일한 확률로 선택되어진다. 즉, 균일 분포(uniform distribution)를 따른다. 패킷 지연 시간은 패킷이 큐에 도착한 시간

과 출력단으로 전송된 시간의 차로 표시하였으며 그 시간은 도착과 전송시의 각각의 타임슬롯 번호를 이용한 타임스탬핑 기법을 사용하였다.

4.2 시뮬레이션 결과 및 분석

본 논문에서 제안하는 수정된 PMRN 스위치에서의 분리된 HOL을 이용한 셀 스케쥴링 기법의 성능을 평가하기 위해서 기존의 PMRN 스위치에서의 결과와 비교한다.

다음은 본 논문에서 제안한 방법의 결과로 멀티캐스트 비율을 증가시켜가면서 서로 다른 윈도우 크기일 때의 최대 입력버퍼의 처리율을 조사한 것이다. 그리고 그림 4.2는 기존의 PMRN 스위치에서의 결과이다. 그림 4.1에서 윈도우 크기가 1일 경우는 입력버퍼의 선두의 셀만이 고려되므로 그림 4.2에서와 같이 기존의 PMRN 스위치에서와 같은 결과를 보인다.

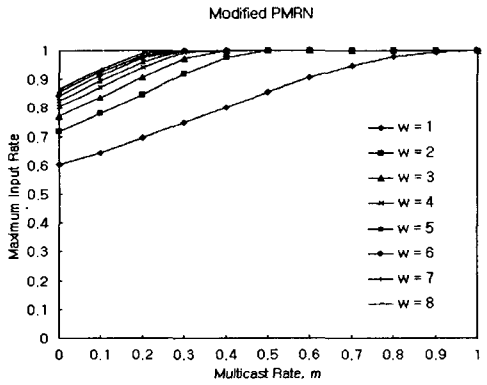


그림 4.1. 수정된 PMRN 스위치에서의 최대 입력 버퍼 처리율

그리고 윈도우 크기가 2일 경우 멀티캐스트 비율이 0.5이상이면 100%의 처리율을 가진다. 또한 윈도우의 크기가 커질수록 더욱 낮은 멀티캐스트 비율에서 처리율이 100%에 도달하므로 기존의 PMRN에서 보다 좋은 성능을 보인다.

수정된 PMRN스위치에서 fanout이 2이고 멀티캐스트 비율이 0.2일 때 윈도우 크기에 따라 입력부하의 증가에 따른 평균 입력 버퍼지연을 그림 4.3에서 보였다.

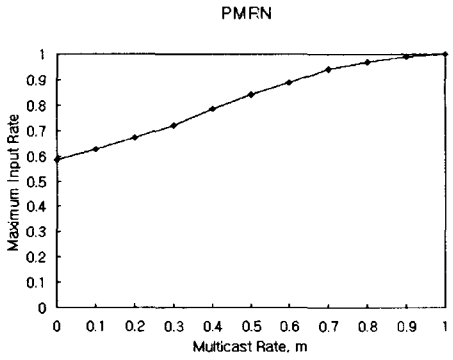


그림 4.2. PMRN 스위치에서의 최대 입력버퍼 처리율

입력 부하가 0.6일 때를 비교해 보면 기존의 방법과 같은 윈도우 크기가 1일때는 289.2(time slots)의 지연을 보인다. 반면에 윈도우 크기가 2이상일 때는 0.5(time slots)정도의 지연을 보임을 알 수 있다.

그림에서 나타난바와 같이 평균 입력버퍼 지연은 같은 입력 부하에서 윈도우 크기가 커질수록 낮아짐을 알 수 있다.

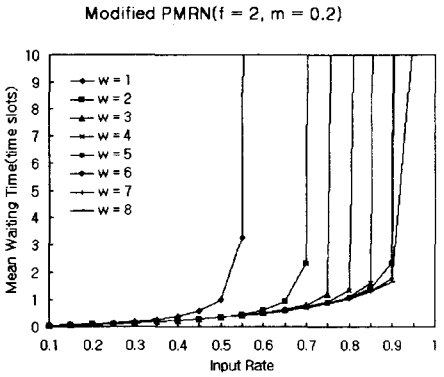


그림 4.3. 수정된 PMRN 스위치에서의 윈도우 크기에 따른 평균 입력버퍼 지연

수정된 PMRN 스위치 상에서 윈도우 크기가 8일 때의 서로 다른 멀티캐스트 비율에서의 입력부하의 증가에 따른 평균 입력버퍼 지연을 그림 4.4에서 보였고, 그림 4.5는 기존의 PMRN 스위치에서의 결과이다.

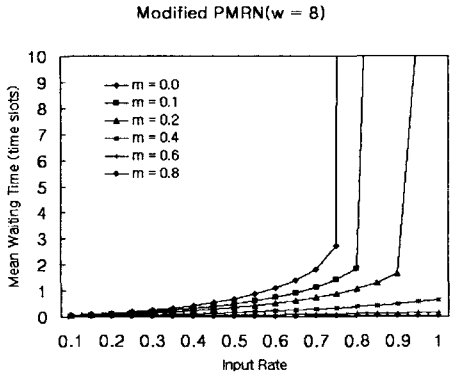


그림 4.4. 수정된 PMRN 스위치에서의 평균 입력 버퍼 지연

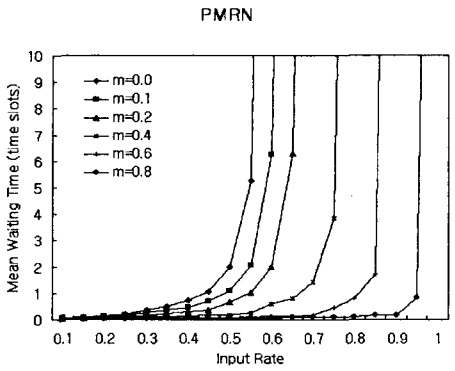


그림 4.5. PNRN 스위치에서의 입력버퍼 지연

그림에서 멀티캐스트 셀의 도착율이 0.2이고 입력 부하가 0.65일 때를 비교해 보면 본 논문에서 제안하는 방법에서는 0.6(time slots)의 지연을 보이고, 기존의 PMRN 스위치에서는 6.3(time slots)의 지연을 보인다. 그림에서 보는 것과 같이 기존의 PMRN 스위치에서 비해 본 논문에서 제안하는 분리된 HOL을 사용하는 기법을 사용함으로써 더 낮은 지연을 보임을 알 수 있다.

각각 멀티캐스트 비율은 0.2이고, fanout이 2와 5일 때 윈도우 크기에 따라 수정된 PMRN 스위치에서의 출력 버퍼의 개수에 따른 최대 처리율을 그림 4.6과 4.7에서 보였다. fanout이 2일 때와 5일 때를 비교해 보면 각각의 그림에서 윈도우 크기가 커질수록 스위치의 처리율이 높아짐을 알 수 있다.

윈도우 크기가 8일 때를 보면 fanout이 2일 때는 출력버퍼의 개수가 8일 때 처리율이 100%에 도달하고, fanout이 5일 때는 출력버퍼의 개수가 2이면 처리율이 100%에 도달한다. 각각의 경우 윈도우의 크기가 1일 때 즉, 입력버퍼의 선두의 셀 만이 고려될 때보다 윈도우 크기가 2로 조금만 늘어도 상당한 처리율의 증가를 보인다. 그러므로 적은 윈도우의 크기로 높은 처리율을 얻을 수 있기 때문에 윈도우로 인한 추가적인 하드웨어 복잡도의 증가는 그리 크지 않을 것이다.

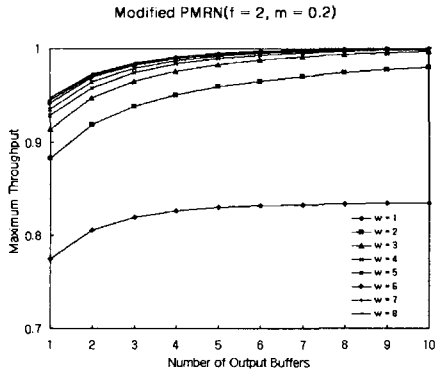


그림 4.6. 수정된 PMRN 스위치에서의 최대 처리율(f = 2)

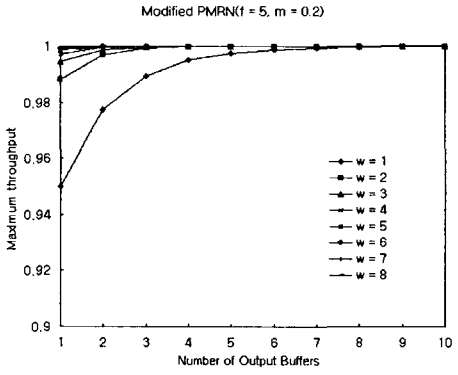


그림 4.7. 수정된 PMRN 스위치에서의 최대 처리율(f = 5)

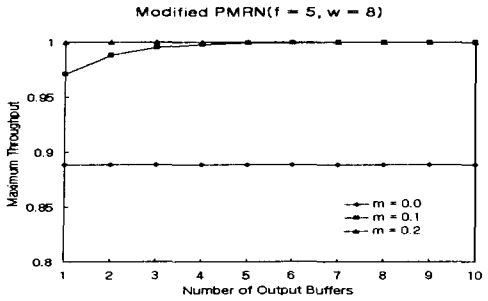


그림 4.8. 수정된 PMRN 스위치에서의 최대 처리율(f = 5, w = 8)

수정된 PMRN 스위치에서 fanout이 5이고 윈도우 크기가 8일 때 서로 다른 멀티캐스트 비율일 때 출력버퍼 크기에 따른 최대 처리율을 그림 4.8에서 보였으며, 그림 4.9에서는 PMRN 스위치에서의 최대 처리율을 보인 것이다. 각각 멀티캐스트 비율이 0.2일 때를 비교해 보면 분리된 HOL을 사용한 방법에서는 최대 처리율이 출력버퍼의 개수가 2일 때 100%에 도달하게 된다. 그리고 PMRN 스위치에서는 출력버퍼의 개수가 5일 때 최대 처리율이 100%에 도달한다.

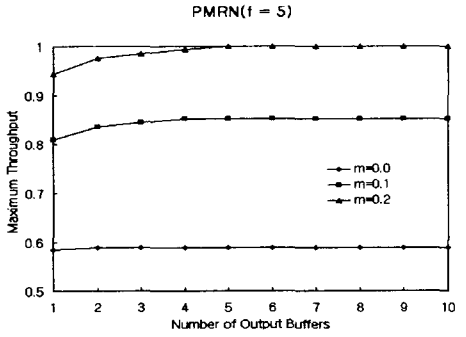


그림 4.9. PMRN 스위치에서의 최대 처리율(f = 5)

수정된 PMRN 스위치에서 fanout이 2이고 윈도우 크기가 8일 때 서로 다른 멀티캐스트 비율에서 출력버퍼 크기에 따른 최대 처리율을 그림 4.10에서 보였으며, 그림 4.11는 PMRN스위치에서의 최대 처리율을 보인 것이다. 앞에서와 같이 각각 멀티캐스트 비율이 0.2일 때를 비교해 보면 분리된 HOL을 사용한 방법에서는 최대 처리율이 출력버퍼의 개수가 8일 때 100%에 도달하게 된다. 그리고 PMRN스위치에서는 출력버퍼의 개수가 3일 때 최대 처리율이 80%에 도달한다.

각각 fanout이 2와 5일 때 멀티캐스트 셀의 도착율이 0% 즉, 유니캐스트 셀만이 들어온다면 기존의 PMRN 스위치에서는 FIFO방식으로 인해 처리율이 0.586으로 제한되고, 본 논문에서 제안하는 방법에서도 윈도우 기법의 한계인 0.88로 처리율이 제한된다. 하지만 실제 상황에 더 가까운 유니캐스트 셀과 멀티캐스트 셀이 혼합된 트래픽에서는 아주 높은 처리율을 보인다. 특히 본 논문에서 제안하는 분리된 HOL을 사용한 기법을 적용했을 때 입력버퍼의 선두에 있는 유니캐스트 셀로 인한 멀티캐스트셀의 전송 방해를 최소화함으로써 기존의 PMRN 스위치 보다 나은 성능을 보임을 시뮬레이션 결과로 알 수 있다.

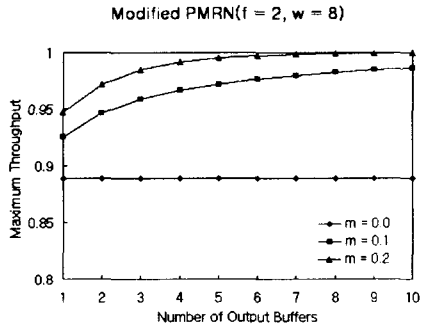


그림 4.10. 수정된 PMRN 스위치에서의 최대 처리율(f = 2, w = 8)

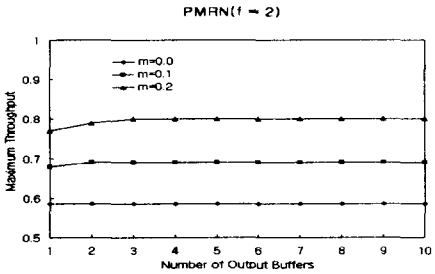


그림 4.11. PMRN 스위치에서의 최대 처리율(f = 2)

수정된 PMRN 스위치에서 fanout이 5이고 윈도우 크기가 8일 때 서로 다른 출력버퍼의 개수에 대해 입력 부하에 따른 스위치전체의 평균 지연을 그림 4.12에서 보였다. 그리고 그림 4.13은 기존의 PMRN 스위치에서의 결과이다.

출력버퍼의 개수가 15일 때를 비교해 보면 입력 부하가 0.5일 때 기존의 PMRN 스위치에서의 지연은 4.1이고 수정된 PMRN 스위치에서는 3.3이다. 그리고 입력 부하가 증가함에 따라 기존의 스위치에서는 지연의 증가가 급격히 일어나지만 수정된 PMRN 스위치에서는 그 증가가 아주 완만함을 알 수 있다. 그러므로 본 논문에서 제안한 기법이 더 나은 성능을 보임을 알 수 있다.

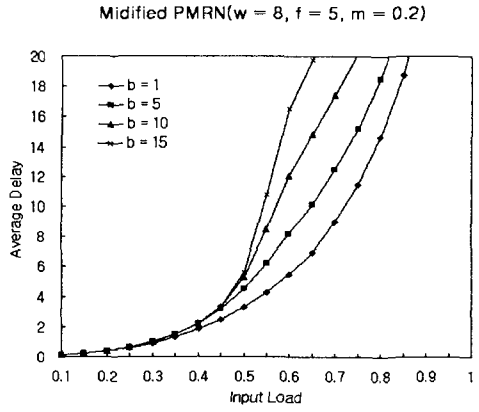


그림 4.12. 수정된 PMRN 스위치에서의 전체 지연(f = 5)

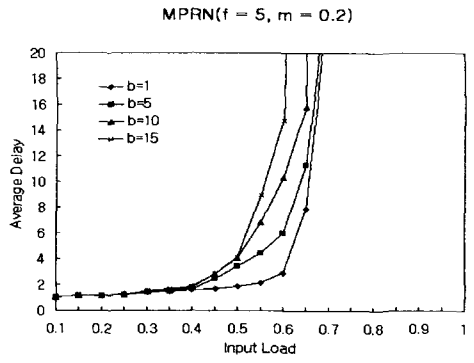


그림 4.13. PMRN 스위치에서의 전체 지연

V. 결 론

본 논문에서는 링망을 이용한 병렬 멀티캐스트 패킷 스위치에서의 분리된 HOL을 사용한 셀 스케줄링 알고리즘을 제안하였다. 분리된 HOL을 사용한 기법은 링망과 일대일 연결망의 앞단에 각각 멀티캐스트 셀과 유니캐스트 셀을 위한 셀 하나 크기의 버퍼를 설정하고 입력버퍼에는 윈도우를 설정하여 윈도우 크기 내에서 셀을 선택할 수 있게 함으로써 멀티캐스트 셀과 유니캐스트 셀이 동시에 선택될 수 있게 하여 스위치의 처리율을 높였다.

시뮬레이션 결과 본 논문에서 제안한 셀 스케줄링 알고리즘을 링망을 이용한 병렬 멀티캐스트 패킷 스위치에 적용함으로써 입력버퍼에서와 전체 스위치에서의 평균 지연은 작은 윈도우 크기에서도 기존의 스위치에 비해 더 낮아짐을 알 수 있다. 또 스위치의 전체 처리율에서도 기존의 방식보다 낮은 멀티캐스트 비율에서 처리율이 쉽게 100%에 도달한다. 그리고 하나의 입력버퍼에서 멀티캐스트 셀과 유니캐스트 셀을 동시에 보낼 수 있어 링망과 일대일 연결망의 이용율을 높일 수 있다.

향후 연구 과제로는 본 논문에서 제안하는 기법의 수학적 분석 모델을 개발하는 것과 스위치에서의 셀 손실율에 대한 성능평가, 그리고 균일한 트래픽이 아닌 버스티(bursty)트래픽과 비 균일(non-uniform) 트래픽에서의 성능을 평가하는 것이 있다.

참고문헌

[1] Walter J. Goralski "Introduction to ATM networking" McGRAW-HILL.

[2] Hamid Ahmadi, and Wolfgang E. Denzel "A Survey of modern high-performance switching techniques", IEEE Journal on Selected Areas Communication, vol. 7, No. 7, pp. 1091-1103, September 1989.

[3] Anthony S. Acampora "An introduction to broadband networks," Plenum press.

[4] Michael G. Hluchyj, and Mark J. Karol "Queueing in high-performance packet switching," IEEE Journal on Selected Areas in

Communication, vol. 6, No. 9, December 1988.

[5] M. J. Karol, M. G. Hluchyj, and S. P. Morgan, "Input versus output queueing on a space-division packet switch," IEEE Trans. on Com. vol. COM-35, No. 12, December 1987.

[6] H. Matsunaga and H. Uematsu, "A 1.5 Gb/s 8x8 cross-connect switch using a time reservation algorithm," IEEE Journal Selected Areas Communication, vol. 9, pp. 1308-1317, 1991.

[7] Balaji Prabhakar, and Nick McKeown, "Designing a multicast switch scheduler," INFOCOM 1996.

[8] Jinchun Kim, Byungho Kim, Hyunsoo Yoon, Jung Wan Cho, "A parallel multicast fast packet switch with ring network and its performance", IEICE Trans. Communication, vol. E79-B, No. 1, January 1996.

김 진 천(Jin-Chun Kim)

1983년 2월 한양대학교 전기공학과(공학사)

1985년 8월 미국 미시간주립대학교 전자 및 시스템공학과(공학석사)

1996년 2월 한국과학기술원 전산학과(공학박사)

1988년 4월 ~ 1993년 5월 삼성종합기술원 정보시스템연구소 선임연구원

1996년 3월 ~ 현재 경성대학교 전기전자 · 컴퓨터공학부 조교수

※ 연구분야 : 멀티미디어 통신, ATM 스위치 구조, 컴퓨터 구조, 초고속 네트워