

# ATM 다중화기에서 셀 스케줄링을 위한 병렬 우선순위 큐잉 알고리즘

유 초 룡, 김 미 영, 권 택 근  
충남대학교 컴퓨터공학과

## Parallel Priority Queuing Algorithm for Cell Scheduling In ATM Multiplexers

Cho-Rong Yu, Mi-Young Kim, Taeck-Geun Kwon  
Dept. of Computer Engineering, Chungnam National University

### 요 약

WFQ(Weighted Fair Queuing)은 지연이나 공평성의 특성에 있어서 이상적인 트래픽 스케줄링 알고리즘으로 간주되었다. N 세션에 서비스를 제공하는 WFQ 스케줄러의 스케줄링 연산은 각 패킷 전송 시간당  $O(n)$ 의 계산 복잡도를 가지며, 구현 또한 복잡하다. Self-Clocked Fair Queuing 과 같은 WFQ 알고리즘의 구현을 간단히 하고자 하는 노력은 지연범위나 특성에 영향을 주게 되어 다양한 트래픽이 제공되는 경우 각 트래픽의 공평성을 지원하지 못한다. 그러므로 지연이나 지연 변이 측면에서 공평성을 지원하고 구현상의 계산 복잡도를 줄인 스케줄링 알고리즘이 필요하게 되었다. ATM 다중화기의 셀 스케줄링 알고리즘 역시, ATM의 특성상 다양한 특성의 서비스를 제공하기 위해서, 다양한 특성의 트래픽에 대한 공평성을 제공하는 새로운 알고리즘의 연구가 필요하다. 이 논문에서는 ATM 스위치 내의 다중화기에서 사용되는 새로운 셀 스케줄링 알고리즘을 제안하고 실험을 통해 이 알고리즘의 성능을 검증하고자 한다. 이 알고리즘은 여러 개의 우선순위 큐를 갖고, 각 우선순위 큐마다 스케줄링 연산이  $O(1)$ 의 계산 복잡도를 갖는 Parallel Priority Queuing 알고리즘이다.

### 1. 서론

현재의 텔레커뮤니케이션 통신망은 데이터나 오디오, 비디오와 같은 다양한 종류의 서비스들을 공동 자원의 공유를 통해 제공해야 한다. ATM (Asynchronous Transfer Mode)는 현재의 텔레 커뮤니케이션에서 요구되는 이런 다양한 종류의 서비스를 제공하는 전송 기술이다. 다양한 종류의 서비스는 서로 다른 종류의 QoS (Quality of Service)를 요구한다. 예를 들어, 데이터 트래픽이 손실에 민감한데 반해, 음성 트래픽은 지연이나 지연 변이에 매우 민감하다. ATM 통신망 안에서 패킷은 고정된 크기의 셀로 나누어진 다음, 고속 링크로 다중화된다. ATM 링크 안에서 셀들의 통계적 다중화는 유동성이나 통신망 자원의 할당 면에서 다른 특성을 가진 트래픽 소스들로부터 이루어진다. ATM 스위치에서는 여러 개의 다중화기를 가지고, 각 다중화기에서는 여러 입력 패킷 중에서 출력 링크로 나갈 패킷을 선택하는 스케줄링 연산을 수행한다. 다양한 트래픽 성질을 고려하지 않은 스케줄링의 경우, 각 트래픽은 셀 지연이나 셀 지연 변이 등에 영향을 받게 되고, 트래픽 사이의 공평성을 저하시키는 결과가 발생한다.[4]

본 논문에서는 ATM 다중화기내에서 트래픽 사이의 공평성을 제공하는 셀 스케줄링 알고리즘을 제안하고자 한다. PPQ (Parallel Priority Queuing)으로 정의되는 이 알고리즘은 각 트래픽마다 우선순위 큐를 할당하여, 가장 우선순위가 높은 패킷을 선택하는 스케줄링 연산을 수행하게 된다. 스케줄링에서 고려된 점은 스케줄링 연산시의 계산 복잡도로, 병렬적으로 수행되는 프로세스를 할당하여 삽입/삭제시의 계산 복잡도를 각 프로세스당  $O(1)$ 으로 감소시켰다. 기존의 스케줄링 알고리즘의 대표적인 Fair Queuing 알고리즘의 계산 복잡도가  $O(\log N)$ 에 이르는 것에 비교할 때 현저한 성능의 향상을 보인다 [2]

논문은 다음과 같은 구성으로 이루어진다. 2 장에서는 ATM 다중화기의 셀 스케줄링 알고리즘에 대한 배경지식에 대한 내용을 기술하고, 3 장에서는 본 논문에서 제안하는 PPQ 알고리즘에 대해서 설명한다. 4 장에서는 PPQ 알고리즘의 실험 및 결과에 대한 내용을 기술하고, 5 장에서는 본 논문의 결론을 맺는다.

### 2. 셀 스케줄링 기법

다양한 특성의 트래픽 서비스를 지원하는 ATM 스위치는 하나의 스위치 패브릭이 여러 개의 다중화기를 가지며, 다양한 트래픽에 대해 각각에 대한 버퍼를 두고 버퍼별로 입력되는 패킷들을 다중화하여 전송될 패킷을 결정하고 전송한다 [7] 다음은 ATM 스위치 모델에 대한 그림이다

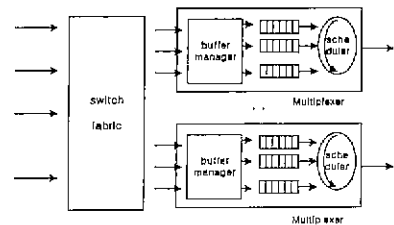


그림 1 ATM 스위치 모델

각 다중화기는 다중 버퍼로부터 입력 받은 패킷에 대한 스케줄링 연산을 수행하여, 다음 사이클에서 출력 링크로 나갈 패킷을 결정한다. 다중화기 모델은 다중 버퍼와 스케줄러로 구성된다 [6] 스케줄러는 다중 버퍼의 패킷들을 스케줄링하여 하나의 출력 패킷을 결정하여 전송한다. ATM 개발초기의 대표적인 셀 스케줄링 알고리즘은 Round-Robin 방식의 Fair Queuing 알고리즘

름이다.

Fair Queuing 알고리즘은 다중 버퍼 모델에서 각 버퍼별로 공평성을 제공하기 위해 Round-robin 방식으로 모든 버퍼를 순차적으로 서비스하는 알고리즘이다. 그러나 Fair Queuing은 트래픽의 우선순위에 관계 없이 공평한 서비스를 제공하기 때문에, ATM에서 지원하는 다양한 트래픽의 특성이 무시되어 공평성을 유지할 수 없다는 단점이 있다 [2] 이런 단점을 보완하기 위해서 WFQ (Weighted Fair Queuing), GPS (Generalized Processor Sharing), BRFQ (Bit-Round Fair Queuing), Frame-based Queuing과 같은 많은 알고리즘들이 나타나게 되었다.[3,4] 다양한 트래픽과 서비스를 제공하는 ATM 스위치의 경우에는 다양한 트래픽 특성에 민감한 스케줄링 알고리즘이 필요하다. ATM 스위치를 기반으로 연구된 알고리즘에는 VC (Virtual Connection)마다 스케줄러를 두고 GPS에 기반한 Per-VC 스케줄링 기법[1], 트래픽의 성질에 따른 클래스기반의 스케줄링 기법에 이르기까지 다양한 스케줄링 알고리즘이 연구되어 왔다. 그러나 연산과정의 계산 복잡도가 복잡하다는 한계점을 가지고 있기 때문에, 계산 복잡도를 줄이면서도, ATM 스위치에서 요구되는 다양한 트래픽에 대한 공평성을 제공하는 새로운 알고리즘에 대한 연구가 필요하게 되었다

### 3. 병렬우선순위큐잉 (Parallel Priority Queuing)

#### 3.1 PPQ 알고리즘의 개요

본 논문에서 제안하고자 하는 ATM 다중화기 내의 셀 스케줄링 알고리즘은 다양한 트래픽 특성에 따른 공평성 보장에 따른 문제점을 극복하기 위하여 제안된 알고리즘이다. PPQ 알고리즘은 ATM 다중화기의 입력 트래픽마다 우선순위 큐를 두고 스케줄링 연산을 통해 각 큐에서 가장 우선순위가 높은 패킷을 결정하여 전송하게 된다. 스케줄링에 사용되는 알고리즘은 자료구조의 힙 구조를 응용한 PPQ에 기반을 두고 있다. PPQ는 일반 최소 힙이나 최대 힙과 같은 구성을 보인다. 힙 구조에서 삽입/삭제 연산의 계산 복잡도인  $O(\log N)$ 에 비해 PPQ의 삽입/삭제 연산의 계산 복잡도는  $O(1)$ 이 된다. PPQ는 이진 트리의 각 레벨 별로 삽입/삭제 연산을 병렬적으로 수행하는 프로세스를 둔다. 다음은 PPQ에서의 삽입/삭제 연산에 대한 예제이다.

##### 1) PPQ에 새로운 노드가 삽입되는 경우

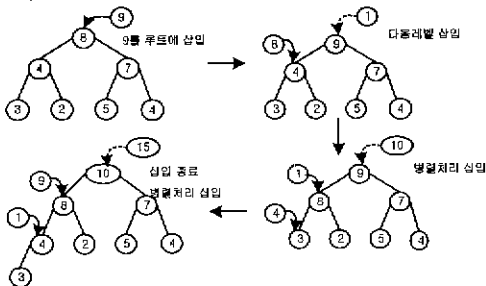


그림 3. 삽입 연산 과정

삽입 연산의 경우 루트 노드에 삽입 연산이 이루어진 후, 새로운 삽입 위치와 삽입 노드가 다음 레벨의 루트 노드로 내려오게 된다. 그림 3에서 PPQ는 최대 힙의 구조를 가지고 있다. 루트 노드의 최대 원소를 우선순위가 가장 높은 원소로 가정한다. 새로운 키값을 가진 노드가 루트에 삽입되면, 루트 노드와 새로운 노드를 비교하여 다음 레벨에 삽입될 삽입 노드를 결정하고, 삽입 위치도 결정한다. 한 레벨에서의 삽입이 완료된 후, 결정된 삽입 위치에 삽입노드를 삽입한다. 삽입 위치

와 삽입 노드를 계산하여, 다음 레벨의 삽입 위치에 삽입 노드에 대한 삽입 연산을 수행한다

##### 2) 병렬우선순위큐에서 우선순위가 가장 높은 노드를 삭제하는 경우

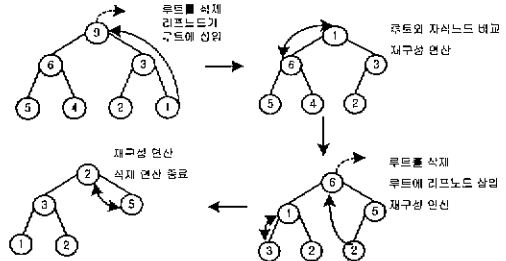


그림 4. 삭제 연산 과정

삭제 연산의 경우도 삽입 연산의 경우와 마찬가지로 루트 노드가 삭제된 후에 새로운 루트를 결정하기 위한 재구성 연산과정이 각 레벨 별로 수행된다. 그림 4에서 보는 바와 같이 루트 노드가 삭제되면 마지막 리프 노드가 루트에 삽입된 후, 자식 노드들과 비교하여 힙의 정의에 위배되지 않도록 PPQ를 재구성한다.

PPQ는 이진 트리의 각 레벨마다 삽입/삭제 연산의 병렬화를 통해서 스케줄링 알고리즘의 계산 복잡도를  $O(1)$ 으로 크게 감소시켰다.

#### 3.2 PPQ의 정의 및 연산

##### □ 정의 1

병렬우선순위큐 (PPQ)는 힙과 논리적으로 동일한 큐로서 삽입/삭제 연산이 각 노드에서 병렬적으로 진행 중이거나 완료된 큐이다

##### □ 정의 2

Q는 병렬우선순위큐 (PPQ)를 나타내고, M\_index는 현재 루트의 인덱스, M\_index'는 다음 삽입 위치에 있는 루트 인덱스, m은 삽입 노드, L\_index는 마지막 리프 노드의 인덱스로 정의한다. 위에 정의된 기호들은 다음의 알고리즘을 표현하는데 사용된다.

##### □ 삽입/삭제 연산의 알고리즘

삽입 함수는 삽입 노드를 루트와 비교하여 다음 삽입 노드를 결정하고, 파라미터로 입력받은 Q\_count와 M\_index를 사용하여 삽입노드의 위치를 결정하게 된다. 병렬우선순위큐는 힙의 구조를 가지고 있기 때문에, M\_index의 값은 삽입 노드가 몇 단계 큐에 삽입될 것인가를 결정, 2의 곱수를 생성하면 그 값은 Q\_count와 나머지 연산에 사용되고 다음 루트를 결정한다.

##### Algorithm Insert(Q\_count, M\_index, m)

- (1) if (M\_index is not greater than Q\_count)
- (2) Q[M\_index] = m,
- (3) Temp = Q[Index];
- (4) Q[Index] = max (temp, m),
- (5) m' = min (temp, m);
- (6) switch ( Q\_count 's mod operation result )
- (7) case : ISLEFT
- (8) M\_index = M\_index\*2;
- (9) case : ISRIGHT
- (10) M\_index = M\_index\*2+1;
- (11) return M\_index, m,

삭제 함수는 파라미터를 통해 넘겨 받은 M\_index 값을 검사하여 루트 노드 삭제와 리프 노드의 루트 삽입을 수행하거나, 히프 정의에 위배되지 않도록 Q[M\_index]를 자식 노드들과 비교하여 교환하는 연산을 수행한다.

**Algorithm Delete(M\_index, Q\_count)**

```

(1) if( M_index is equal to 1 )
(2)   transn(Q[index]),
(3)   Q[M_index] = Q[L_index];
(4) M_index' = max {Q[M_index], Q[M_index*2],
                   Q[M_index*2+1]},
(5) if( M_index' is not equal M_index )
(6)   Swap(Q[M_index], Q[M_index']);
(7)   if(M_index' is equal to M_index*2 )
(8)     M_index = M_index*2;
(9)   else
(10)    M_index = M_index*2+1,
(11) return M_index';
    
```

**4. 실험 및 결과**

**4.1 실험모델**

PPQ 알고리즘의 성능을 실험하기 위한 시스템 모델은 다음과 같다.

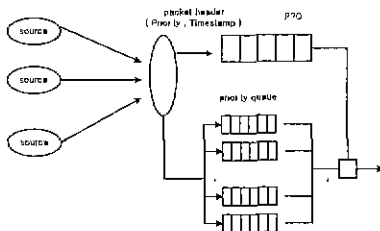


그림 5. 시스템 모델

본 논문에서 실험 할 시스템 모델은 N 개의 트래픽 소스로부터 생성된 패킷을 우선순위 큐에 저장하였다가, PPQ 알고리즘을 통한 스케줄링의 결과를 통해 출력되는 패킷을 결정하여, 전송하는 모델이다. 실험에 사용되는 파라미터들은 패킷 헤더 부분인 우선순위와 타임스탬프이다. 각 패킷이 가진 우선순위를 통해 PPQ 알고리즘을 수행하고, 타임스탬프를 통해 스케줄링에 의해 발생하는 지연 시간을 측정한다.

**4.2 실험조건**

실험에서 성능비교를 위해서 사용되는 알고리즘은 FIFO 이고, 트래픽 소스가 패킷을 생성하여 PPQ로 전송하는 시간과 순서는 임의함수를 사용하여, 균등하게 배분되도록 하였다.

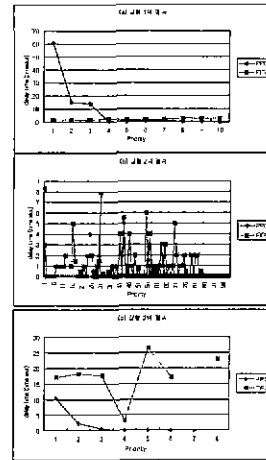
실험 1에서는 트래픽 소스가 10 개이고, PPQ에 대한 각 트래픽 소스의 부하가 균등한 경우, 실험 2에서는 트래픽 소스가 100 개이고, PPQ에 대한 각 트래픽 소스의 부하가 균등한 경우, 실험 3에서는 트래픽 소스가 10 개이고, PPQ에 대한 각 트래픽 소스의 부하가 불균등한 경우 등 세가지 경우의 지연시간의 변화를 측정하였다. 실험 3의 불균등한 트래픽 부하의 실험은 낮은 우선순위 트래픽에 부하가 가중되는 경우를 실험하였다.

**4.3 실험결과**

그림 6 (a), (b)의 결과는 우선순위가 높은 패킷 지연시간이 현저하게 감소함을 나타낸다. 트래픽 소스의 개수를 다르게

한 경우도 같은 결과를 보인다. 그림 6 (c)의 결과 역시 트래픽 부하의 변화에 영향 받지 않고, 같은 결과를 나타낸다. 그러므로, PPQ 알고리즘은 트래픽 부하나 트래픽 수의 상태 변화에 영향을 받지 않고 우선순위 트래픽의 지연시간을 감소시키는 결과를 얻을 수 있다.

그림 6 실험 결과



**5. 결론**

본 논문은 다양한 특성의 트래픽에 대한 공평성을 제공하고, 연산의 계산 복잡도를 감소시킨 ATM 다중화기의 셀 스케줄링 알고리즘인 PPQ에 대한 연구 내용을 기술하고 이 알고리즘의 성능을 실험을 통해서 평가하였다. PPQ는 삽입/삭제 연산의 계산 복잡도를 줄이고, 각 트래픽의 특성에 따른 우선순위를 고려한 알고리즘이다. 본 논문은 PPQ의 성능을 평가하기 위해 각 패킷의 스케줄링에 소요되는 지연 시간을 측정하였다. 실험 결과, 우선순위에 따른 지연 시간의 단축이 검증되었다.

**참고 문헌**

- [1] Uwe Briem, Eugen Wallmeier, "Traffic Management for an ATM Wswitch with Per-VC Queuing: Concept and Implementation", IEEE Communication Magazine, pp. 88-93, January 1998.
- [2] M.Shreedhar, G.Varghese, "Efficient Fair Queuing using Deficit Round Robin", Proc. SIGCOMM'95, September 1995.
- [3] Dimitrios Stiliadis, Anujan Varma, "Rate-Proportional Servers. A Design Methodology for Fair Queuing Algorithms", IEEE/ACM Transaction on Networking, April 1996
- [4] Dimitrios Stihadis, Anujan Varma, "Efficient Fair-Queuing Algorithms for ATM and Packet Networks". IEEE INFOCOM'96, pp. 111-119, March 1996.
- [5] M. O. Lai, Danny H. K Tsang, "Modified Fair Queuing for Finite Buffer in ATM Networks", Proc. IEEE INFOCOM'99, pp 193-194, 1999.
- [6] Reza Jafri, Khosrow Sohraby, "Performance Analysis of a Priority based ATM Multiplexer with Correlated Arrival", Pro IEEE INFOCOM'99, pp 1036-1037, 1999
- [7] M Katevenis, S. Sidiropoulos, "Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip", IEEE Journal on Selected Areas in Communications, Vol.9, p 1265-79, October 1991