

평탄화 기법을 개선한 WRR 셀 스케줄링 알고리즘

WRR Cell Scheduling Algorithm for Improving Smoothing Scheme

조해성
건양대학교 IT학부

Hae-Seong Cho (hscho@konyang.ac.kr)
School of IT, Konyang University

중심어 : 평탄화, WRR, 스케줄링 알고리즘, 멀티미디어 서비스

Keyword : Smoothing Scheme, WRR, Scheduling Algorithm, Multimedia Service

요 약

본 논문에서는 기존의 Smoothing Scheme의 문제점을 개선한 새로운 WRR 셀 스케줄링 알고리즘을 제안한다. 제안된 셀 스케줄링 알고리즘에서는 현재 라운드에 서비스되는 VC의 큐가 비어있을 경우 모든 VC들 중에서 할당된 가중치와 버퍼의 길이의 차이가 가장 큰 VC를 서비스한다. 이렇게 함으로써 대역 할당의 효율이 증가하는 결과를 가져온다. 또한, 모든 실시간 트래픽을 서비스한 후에 남은 멀티미디어 서비스 대역을 비 실시간 트래픽에 할당할 수 있다. 한 사이클 안에 평균 서비스 수가 증가하고 많은 셀을 서비스하게 됨으로써 총 버퍼 길이도 감소한다.

Abstract

In this paper, I proposed a new WRR Cell Scheduling algorithm for improving the problem of existing smoothing scheme. In proposed algorithm, when queue of being service VC in present round is emptied, instead VC that difference between assigned weight and length of present buffer is biggest in all VC service. This result efficiency of bandwidth allocation is increased. And above all real-time traffic is serviced, non-real-time to allocate bandwidth of the remainder multimedia-service is possible. In one cycle, number of serviced average cell is increased. And for many cell serviced, length of total buffer is decreased.

I. 서론

트래픽 제어 방법들 중 ATM 망 자원의 효율적인 사용을 위하여 연구되고 있는 분야가 트래픽 흐름 제어이다. 트래픽 흐름 제어를 위해서는 각 스위치의 출력 단에서 각 VC (Virtual Connection)에 공평한 대역을 할당할 수 있어야 한다. 이를 수행하기 위해서는 셀 스케줄링 알고리즘이 필수적이다. 셀 스케줄링의 성능을 결정하는 요소들은 트래픽의 통계적 특성, QoS(Quality of Service) 파라미터, 스케줄링 알고리즘이다. 이 중 스케줄링 알고리즘은 셀 스케줄링의 성능을 결정하는 가장 중요한 요소이다. 셀 스케줄링 알고리즘이 고려해야 할 사항은 트래픽 흐름의 분리, 셀 지연, 자원 이용률, 공평성, 알고리즘의 간편성, 알고리즘의 확장용이성 등이다. 지금까지 연구되어온 셀 스케줄링 알고리즘의 일반적인 특징을 요약하면 다음과 같다. 첫째 스케줄러는 셀 소스에 일정한 전송률로 협상한다고 가정한다. 그러므로 대부분의

스케줄러는 각 셀 소스가 협상된 전송률을 위반하지 않는 한 일정한 성능을 보장하는 방식으로 구성된다. 둘째는 대부분의 스케줄러는 각각의 가상 연결에 하나의 큐가 할당되는 per-connection 큐잉을 사용한다. 또한, 대부분의 셀 스케줄러들은 작업 보존형(work conserving)과 비 작업 보존형(non work conserving)의 두 가지로 분류되어 있다[1].

스케줄링 방법에는 지연 특성을 고려한 스케줄링 방법과, 대역폭 할당을 기본으로 하는 스케줄링 방법, 그리고 우선순위를 고려한 스케줄링 방법 등으로 나눌 수 있다. 첫 번째 방법의 대표적인 것으로 Deadline 스케줄링이 있다[2],[3]. 이 방법에서는 Deadline이 가장 촉박한 셀을 우선적으로 처리함으로써 QoS를 제공하고 폐기되는 셀의 수를 최소화한다. 두 번째 방법의 대표적인 것으로 WRR(Weighted Round Robin) 방법이 있다[4]. 싸이클(cycle)이라고 하는 고정된 시간동안 각 소스에 할당된 가중치(Weight)에 따라서 서비스를 제공하는 방법이다. 이를 개선한 방법으로 많이 알려진 것이 동적으로 가중치를 할당하는 DWRR(Dynamic Weighted Round

Robin)이다[5]. 세 번째 방법에는 다시 공간 우선 순위와 시간 우선 순위의 두 가지 방법이 있다. 그리고 시간 우선 순위 방법은 다시 고정(Static or Fixed) 우선 순위 방법과 동적 우선 순위 방법(DPS : Dynamic Priority Scheduling)으로 나눌 수 있다. 대표적인 것으로는 QLT(Queue Length Threshold), MLT(Minimum Laxity Threshold), DMLT(Double MLT) 등이 있다[6].

트래픽 정형화(shaping)는 트래픽 특성들을 원하는 형태로 변경하기 위하여, VCC 또는 VPC상의 셀들의 흐름에 대한 트래픽 특성을 바꾸는 기법이다. 기존의 WRR 스케줄링 기법 중의 하나인 Smoothing Scheme은 셀 손실보다는 지연에 민감한 서비스 클래스인 CBR과 rt-VBR 트래픽에 대해 초점을 맞춘 WRR 스케줄링 알고리즘으로 멀티미디어 서비스를 함께 처리하는 경우 심각한 성능 저하를 일으키며, 셀 스케줄링의 중요한 선행 조건 중에 하나인 버퍼 관리 측면을 간과하여 버퍼 대기 지연, 셀 손실, 버퍼의 낭비 등의 문제들을 초래하게 된다[7].

본 논문에서는 기존의 Smoothing Scheme의 문제점을 개선한 새로운 WRR 셀 스케줄링 알고리즘을 제안한다. 제안된 셀 스케줄링 알고리즘에서는 각 VC의 큐에 실제로 들어온 셀의 개수를 표시하는 QL_i 와 가중치를 나타내는 W_i , 큐 상태를 나타내는 $Diff_i$ ($QL_i - W_i$)의 추가적인 변수를 사용하여 가중치를 초과하여 셀이 입력된 버퍼와 가중치보다 셀이 적게 입력된 버퍼의 VC를 서비스할 수 있도록 하고 실시간 트래픽에 할당하고 남은 대역을 비실시간 트래픽에 제공함으로써 멀티미디어 서비스가 가능하도록 하여 서비스되는 평균 셀의 개수가 증가하고 버퍼의 길이가 감소하는 효과를 얻게 된다.

II. BSM(Better Spreading Method)

최근에 M. Katevenis는 회전식 스캔 개념을 토대로 WRR 셀 스케줄링 기법을 연구하였다[8]. M. Katevenis는 한 VC의 서비스되는 연속된 두 셀 사이의 delay 지터를 최소화하기 위해서 N개의 round로 구성된 cycle 동안에 가중치가 W인 VC의 서비스가 이루어지는 각각의 round를 아래의 식과 같이 할당하였는데, 이를 BSM(Better Spreading Method)이라고 한다.

$$BSM: \left\{ \left\lfloor \frac{N}{W} \right\rfloor, 2 \left\lfloor \frac{N}{W} \right\rfloor, \dots, W \left\lfloor \frac{N}{W} \right\rfloor \right\} \quad (1)$$

$\lfloor N/W \rfloor$: Integer part of N/W

N/W : SG(Spreading Gap)

N : no. of round cycles, W = weight of VC

이러한 BSM은 한 주기의 cycle에서 각 VC의 서비스되는 연속된 두 셀을 같은 간격으로 배분하여 delay 지터를 최소화 할 수가 있다. 그러나 SG 값이 정수가 아닌 경우 BSM은 한 싸이클의 round를 고르고 균등하게 할당하지 못하는 즉, 어떤 round에는 셀이 할당되지 않으며 특정 round에 많은 셀이 집중되어 할당되는 문제점을 갖게된다. 이러한 문제점을 예를 들어 살펴보겠다.

만약 N = 16이고 VC a, b, c, d, e의 가중치는 각각 8이며 VC f, g, h, i, j의 가중치는 각각 9인 경우, Katevenis의 스케줄링 기법을 적용하면 각 VC에 할당되는 round의 집합을 식 1에 의해서 다음과 같이 얻게 된다.

$$VCa=VCb=VCc=VCd=VCE=\{2, 4, 6, 8, 10, 12, 14, 16\}$$

$$VCf=VCg=VCh=VCi=VCj=\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

그림 1은 위의 예제에 대한 서비스 분포를 보여주고 있다. 그림 1에서 보는 것과 같이 VC j의 연속적인 두 셀들은 9, 4, 24 셀 타임 간격으로 떨어져 분포 되어있으며 11, 13, 15 round cycle에서는 한 개도 셀도 서비스가 이루어지지 않고 있음을 알 수 있다. 그래서 이러한 기법의 WRR 스케줄링 기

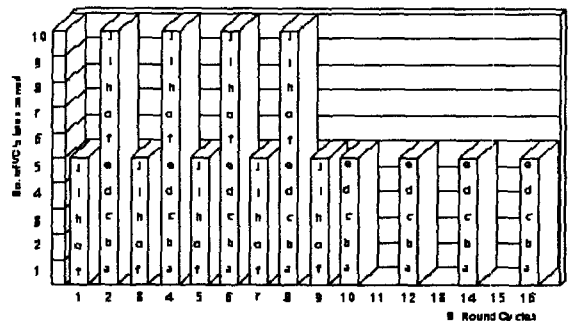


그림 1. BSM에 대한 서비스 분포

법은 몇몇 VC에 대해서 보다 큰 타임 스탬프 지터 문제를 야기시키며, 서비스가 이루어지지 않은 round cycle의 존재로 효율적인 대역 할당에 낭비가 생긴다. 또한 시간적인 측면에서도 전체 VC에 대한 공평한 할당이 수행되지 않는 문제점을 초래하게 된다.

III. Smoothing Scheme

BSM의 스케줄링 기법에서 SG(N/W)의 값이 정수인 경우에는 각 VC의 연속된 두 개의 셀은 같은 간격으로 한 사이클에서 서비스가 이루어지므로 delay의 편차는 0이 된다. 그러나 SG의 값이 정수가 아닌 경우 전체 VC에 대한 시간적인 지연의 공평성을 보장할 수가 없게 된다. 그래서 SG의 값이 정수가 아닌 경우 지연의 편차를 완화하기 위한 기법이 제안되었는데 이를 Smoothing Scheme이라고 한다. 이 기법의 기본적인 원리는 SG(Spreading Gap)의 값이 정수가 아닌 경우 $\lfloor N/W \rfloor$ 와 $\lfloor N/W \rfloor + 1$ 의 두 값을 조합하여, 각 VC에 할당된 round cycle를 전체 round cycle에 전체적으로 균일하게 배치하는 방법을 취하고 있다. 그림 1에서 VC j를 예를 들어 살펴보면 $\lfloor N/W \rfloor = 1$, $\lfloor N/W \rfloor + 1 = 2$ 가 되는데 9개의 셀 중에서 2 round 간격으로 7개의 셀을 서비스하고 1 round 간격으로 2개의 셀을 서비스하게 되는데, 한 사이클에서 지연에 대한 편차를 완화시키기 위해서 이 두 개의 간격을 서로 사이에 끼워서 round를 할당하게 된다. 다음은 Smoothing Scheme에서 사용되는 파라미터들이다.

- N : no. of round cycles, W = Weight of VC
- GAP:= $\lfloor N/W \rfloor$ * Integer part of N/W *
- m = N-1-GAP*W
- F : Flat index of service distribution
- E(k) : Equal index of spreading gap for VC k
- U_F : The average served amounts of cells for a round cycle; [cells/cycle]
- $U_E(k)$: The average spreading gap between two services for VC k with w visits within N cycles; [cycles]
- S_F : Standard deviation of service distribution; [cells/cycle]
- $S_E(k)$: Standard deviation of spreading gap for VC k; [cycles]

$$\text{Fairness}(x_1, x_2, \dots, x_n) = \frac{\left[\sum_{i=1}^n x_i \right]^2}{n \sum_{i=1}^n x_i^2} = \frac{1}{n} \sum_{i=1}^n \frac{x_i}{x_i^2}$$

다음은 그림 1과 그림 2에 대하여 BSM과 Smoothing Scheme에 대한 파라미터 값을 비교한 결과이다.

▶ BSM

- $U_F = (8*5+9*5)/16=5.3$ [cells/cycle]
- $U_E(e) = 16/8=2$ [cycles]
- $U_E(j) = (8*1+1*8)/9=1.8$ [cycles]
- $S_F^2 = \frac{[9*(5-5.3)^2+4*(10-5.3)^2+3*(0-5.3)^2]}{16} = 10.84$
- $S_E^2(e) = 0$ [cycles]
- $S_E^2(j) = \frac{[8*(1-1.8)^2+1*(8-1.8)^2]}{9} = 4.84$ [cycles]
- $F = \frac{[4*10+9*5]^2}{16*[4*10^2+9*5^2]} = 0.72$
- $x_f = \frac{4*10^2+9*5^2}{4*10+9*5} = 7.35$
- $E(e) = \frac{(8*2)^2}{8*8*2^2} = 1$
- $E(j) = \frac{[8*1+1*8]^2}{9[8*1^2+1*8^2]} = 0.40$

▶ Smoothing Scheme

- $U_F = (8*5+9*5)/16=5.3$ [cells/cycle]
- $U_E(e) = 2*8/8=2$ [cycles]
- $U_E(j) = (2*7+1*2)/9=1.78$ [cycles]
- $S_F^2 = \frac{[13*(5-5.3)^2+2*(10-5.3)^2+1*(0-5.3)^2]}{16} = 4.59$
- $S_E^2(e) = 0$ [cycles]
- $S_E^2(j) = \frac{[2*(1-1.8)^2+7*(2-1.8)^2]}{9} = 0.17$ [cycles]
- $F = \frac{[2*10+13*5]^2}{16*[2*10^2+13*5^2]} = 0.86$
- $x_f = \frac{2*10^2+13*5^2}{2*10+13*5} = 6.18$
- $E(e) = \frac{(8*2)^2}{8*8*2^2} = 1$
- $E(j) = \frac{[2*1+7*2]^2}{9[2*1^2+7*2^2]} = 0.95$

Smoothing Scheme은 SG(Spreading Gap)의 값이 정수가 아닌 경우 지연의 편차를 완화하기 위한 기법으로 각 VC의 가중치의 따라 서로 다른 4가지의 분포로 사이클을 할당하게 된다. 첫째, N/W값이 정수인 경우, 이 경우에 해당하는 VC의

셀들은 한 싸이클에서 서로 NW 만큼 떨어져서 분포하게 된다. 둘째, $2N/3 < W < N$ 인 경우, 이러한 조건에 해당하는 VC의 셀들은 2 round 간격으로 $(N-W)$ 개의 셀을 분포시키고 1 round 간격으로 나머지 셀을 분포시킨다. 셋째, $N/2 < W < 2N/3$ 인 경우, 이러한 조건에 해당하는 VC의 셀들은 1 round 간격으로 $(N-W)$ 개의 셀을 분포시키고 2 round 간격으로 나머지 셀을 분포시킨다. 넷째, $1 < W < N/2$ 인 경우, 이 조건을 만족하는 VC의 셀들은 $(GAP + 1)$ round 간격으로 m 개의 셀을 분포시키고 GAP round 간격으로 나머지 셀들을 분포시킨다.

그림 1에 대한 예제에 Smoothing Scheme을 적용시킬 경우 VC a, b, c, d, e는 $SG(=2)$ 값이 정수가 되어 이 VC들의 셀들은 2 round씩 떨어져 분포하게 되며, 나머지 VC들은 Smoothing Scheme 알고리즘의 세 번째 경우에 해당되어 10개의 VC에 대한 서비스 분포가 아래와 같이 얻어진다.

$$VCa=VCb=VCc=VCd=VGe=\{spread\ as\ 2,\ 2,\ 2,\ 2,\ 2,\ 2,\ 2\}$$

$$VCh=VCg=VCh=VCi=VCj=\{spread\ as\ 2,\ 1,\ 2,\ 1,\ 2,\ 2,\ 2,\ 2\}$$

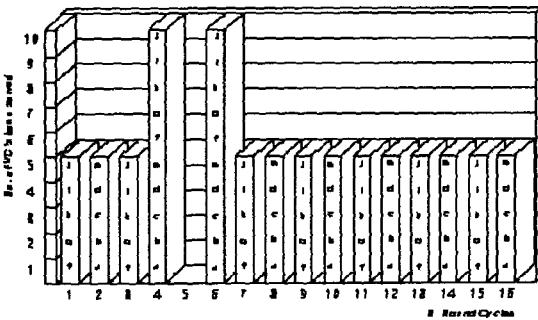


그림 2. Smoothing Scheme에 대한 서비스 분포

Smoothing Scheme은 그림 2에서 보는 것과 같이 BSM에 비해 각 VC의 셀들을 전체 round에 대체적으로 균등하게 분포시켜 지연 편차를 줄이고, 뿐만 아니라 전체 round를 고르게 사용함으로써 대역 할당 측면에서도 우수함을 보이고 있다. 그런데 Smoothing Scheme은 오직 트래픽의 지연 편차만을 개선한 알고리즘으로 이러한 기법은 실시간 특성을 위한 지연 관련 파라미터인 CDV(Cell Delay Variation)와 CTD(Cell Transfer Delay)가 정의되어 있는 CBR이나 rt-VBR 트래픽만을 서비스하는데, 우수한 셀 스케줄링 방법 중에 하나가 될 것이다. 하지만 ATM은 효율적인 대역폭 할당, 다양한 서비스 수용성 등의 장점 때문에 멀티미디어 트래픽에 적합하다고 알려져 있다. ATM Forum에서는 다양한 서비스들

에 대해서 다섯 개의 서비스 클래스를 정의하고, 이들 서비스 클래스들을 모두 지원해야 한다고 규정하고 있는데, Smoothing Scheme은 CBR 트래픽과 rt-VBR 트래픽을 제외한 nrt-VBR, ABR, UBR 트래픽의 서비스 품질을 만족시키는데 문제점을 내포하고 있다. 버스티한 특성을 가지고 있는 nrt-VBR 트래픽이나 ABR 트래픽의 QoS 만족하기 위해서는 버퍼 상태를 고려한 스케줄링 기법이 요구되는데, 만약 버퍼가 가득 차 있는 경우에 각 VC에 할당된 가중치만큼의 셀만을 서비스한다면 그 VC의 버퍼에 계속해서 셀들이 축적되어 버퍼 대기 지연이 커지며 최악의 경우에는 오버플로 현상이 발생하여 셀을 폐기할 수밖에 없는 상황에 이르게 된다.

IV. 제안된 스케줄링 알고리즘

기존의 BSM과 Smoothing Scheme은 셀 손실보다는 지연에 민감한 서비스 클래스인 CBR과 rt-VBR 트래픽에 대해 초점을 맞춘 WRR 스케줄링 알고리즘으로, 멀티미디어 서비스를 함께 처리하는 경우 심각한 성능 저하를 일으킬 것이며, 셀 스케줄링의 중요한 선행 조건 중에 하나인 버퍼 관리 측면을 간과하여 버퍼 대기 지연, 셀 손실, 버퍼의 낭비 등의 문제들을 초래하게 된다.

이 장에서는 기존의 Smoothing Scheme의 문제점을 개선한 새로운 WRR 셀 스케줄링 알고리즘을 제안하였다. 제안된 셀 스케줄링 알고리즘에서는 각 VC의 큐에 실제로 들어온 셀의 개수를 표시하는 QL_i 와 가중치를 나타내는 W_i , 큐 상태를 나타내는 $Diff_i(QL_i - W_i)$ 의 추가적인 변수를 사용하였고, 가중치를 초과하여 셀이 입력된 버퍼와 가중치보다 셀이 적게 입력된 버퍼의 VC를 서비스할 수 있도록 하였다. 서비스는 Smoothing Scheme의 Constant Delay Scheduling Algorithm에 의해서 각 round에 할당된 VC들의 목록을 전체 VC들에 대한 상태정보를 담고 있는 함수인 Variable_Table ($QL_i, W_i, Diff_i$)과 상호 비교하면서 재구성하게 된다. 제안된 알고리즘에서 사용되고 있는 함수 Variable_Table은 각 VC의 버퍼 길이와 할당된 가중치에 대한 정보를 저장하고 있으며, 한 싸이클의 매 round마다 정보를 갱신하게 된다.

Smoothing Scheme과 제안된 셀 스케줄링 알고리즘에서 32개의 round에 각각 VC를 할당할 때 몇 개의 round에 VC들이 할당되지 않는 경우가 발생하는데, 이러한 서비스가 이루어지지 않는 빈 round를 처리하지 않는 경우 대역 할당 분

포의 편차가 심하게 되고 대역 할당의 낭비를 초래하게 된다. 이러한 경우 ABR 트래픽에 우선 순위를 두고 ABR 트래픽이 없는 경우 UBR 트래픽을 처리하기 위한 round로 할당함으로써 실시간 서비스를 지원하는 네트워크에서 다양한 서비스를 지원하도록 하였다. 기존의 알고리즘은 CBR 트래픽과 rt-VBR 트래픽을 제외한 nrt-VBR, ABR, UBR 트래픽의 서비스 품질을 만족시키는데 문제점을 내포하고 있다. 버스티한 특성을 가지고 있는 nrt-VBR 트래픽이나 ABR 트래픽의 QoS를 만족하기 위해서는 버퍼 상태를 고려한 스케줄링 기법이 요구되는데 만약 버퍼가 가득 차 있는 경우, 각 VC에 할당된 가중치만큼의 셀만을 서비스한다면 그 VC의 버퍼에 계속해서 셀들이 축적되어 버퍼 대기 지연이 커지며 최악의 경우에는 오버플로 현상이 발생하여 셀을 폐기할 수밖에 없는 상황에 이르게 된다. 이렇게 트래픽에 대한 지연의 편차를 최소화하는데 주력한 스케줄링 기법은 지연에 민감한 실시간 트래픽인 CBR과 rt-VBR 트래픽을 처리하는데 우수한 성능을 보이지만 nrt-VBR과 ABR, UBR 트래픽을 포함한 멀티미디어 서비스 처리에 대한 미흡한 문제점과 각 VC의 큐 상태에 관계없이 서비스를 수행하기 때문에 버퍼 대기 지연 및 셀 손실을 초래할 수가 있는 문제점을 이 알고리즘으로 개선하였다.

다음은 제안된 알고리즘에서 사용된 함수와 변수에 대한 설명이다.

- ▶ QL_i : VC_i 의 큐의 길이
- ▶ W_i : VC_i 의 가중치 ($1 \leq W_i \leq 32$)
- ▶ VC_i : Constant Delay Scheduling Algorithm()에 의해서 각각의 round에서 서비스를 받게 되는 VC
- ▶ VC_j : 현재 할당되고 있는 round에서 서비스 받지 않는 VC
- ▶ Search() : VC_j 를 검색하는 함수
- ▶ Serve() : VC_i 또는 VC_j 을 서비스하는 함수
- ▶ Variable_Table($QL_i, W_i, Diff_i$) : 각 VC의 큐의 길이, 가중치, ($QL_i - W_i$)에 대한 정보 테이블
- ▶ Service_Table($Round_k, VC_i, VC_j$) : 전체 round에 각각 할당된 VC에 대한 정보 테이블

위에서 설명한 변수와 함수들을 이용하여 제안된 WRR 셀

스케줄링 알고리즘은 다음과 같은 Pseudo-code로 나타낼 수 있다. 다음에 기술하는 Pseudo-code에서 Variable_Table은 매 round마다 갱신되며 Service_Table은 한 사이클마다 갱신되어 한 사이클의 전체 round에 할당할 VC들을 미리 선정하여 서비스가 수행된다.

◆ Proposed Cell Scheduling Algorithm()

```

WHILE(Each  $VC_i$  is allocated in Kth round of the Service_Table)
  IF( $VC_i$  queue is empty)
    search Max[  $QL_j - W_j$  ]  $VC_j$ 
    IF(Max[  $QL_j - W_j$  ]  $\neq$  0)
      Allocate  $VC_j$  in the Kth round
       $QL_j = QL_j - 1$ 
    ELSE(Max[  $QL_j - W_j$  ] = 0)
      IF(There is a ABR source)
        Allocate ABR source in the Kth round
      ELSE
        Allocate UBR source in the Kth round
      END
    END
  ELSE(  $VC_i$  queue is not empty)
    Allocate  $VC_i$  in the Kth round
  END
END
Update Variable_Table(  $QL_i, W_i, Diff_i$  )
K=K+1
IF(There is no allocated  $VC_i$  in the Kth round)
  IF(There is a ABR source)
    Allocate ABR source in the Kth round
  ELSE
    Allocate UBR source in the Kth round
  END
END
END
Call setup
Update the Service_Table(  $Round_k, VC_i, VC_j$  )
    
```

V. 시뮬레이션 및 성능평가

1. 시뮬레이션 환경

시뮬레이션 환경에서 출력 링크의 용량은 128M이고 실시간 트래픽에 할당된 대역과 VC의 개수는 입력 부하에 따라 변화를 주었으며, 비실시간 트래픽에 대해서는 이용 가능한

남은 대역의 30%를 할당하였다. 표 1은 입력 부하에 따른 실시간 트래픽의 개수와 할당 대역을 보여주고 있다.

표 1. 입력 부하와 대역에 따른 실시간 트래픽의 개수

부하 전송률	50%	70%	80%	95%
8M	0	4	4	4
4M	4	4	4	8
2M	8	8	8	8
1M	16	8	16	16
0.5M	16	16	16	16
256K	16	16	16	16
128K	16	16	16	16
64K	32	32	32	32

그림 3에서 보는 것과 같이 시뮬레이션 모델은 N개의 입력 소스와 매 타임 슬롯마다 특정 입력 소스를 선택하여 서비스하는 스케줄러, 스케줄러에 의해서 선택된 셀들이 출력 링크로 빠져나가기 전에 잠시 대기하는 출력 버퍼로 구성되어 있다. 스케줄러는 WRR(Weighted Round Robin) 셀 스케줄링 방식으로 동작하며, Variable_Table에 의해 미리 작성된 Service_Table을 참조하여 서비스를 수행하게 된다.

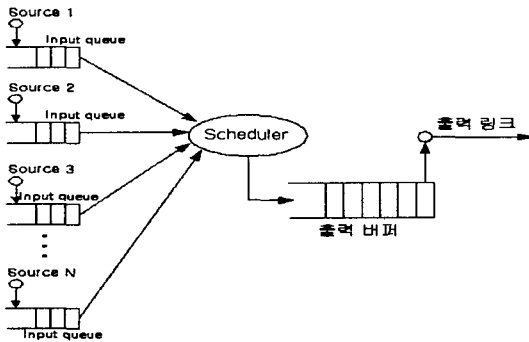


그림 3. 시뮬레이션 모델

2. 시뮬레이션 결과

본 절에서는 표 1의 시뮬레이션 환경에서 시뮬레이션을 수행하여 기존 알고리즘과 제안된 알고리즘을 서비스되는 평균 셀 수, 최대 버퍼의 크기, 대역 할당의 공평성, 서비스 편차의 측면에서 그 결과를 실시간 트래픽만 존재하는 경우와 비실시간 트래픽이 공존하는 두 개의 상황에서 비교, 분석하였다. 그리고 CBR 트래픽과 UBR 트래픽은 시뮬레이션 환경에서 제외가 되었는데, 그 이유는 전자에 대해서 완벽한 QoS

가 보장된다고 가정했기 때문이고, 후자에 대해서는 망 상황에 따라 어떠한 QoS도 보장될 수 없기 때문이다.

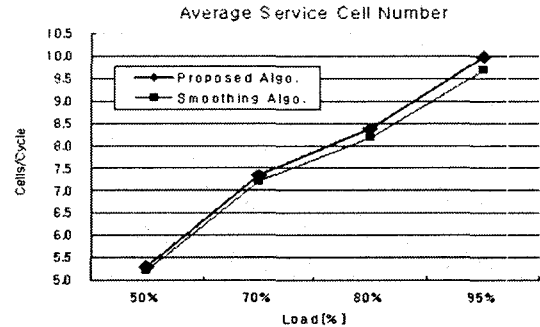


그림 4. 부하에 따른 U_F 값(실시간 트래픽)

그림 4는 실시간 트래픽만을 서비스하는 경우 파라미터 U_F 의 값을 비교한 그래프이다. 파라미터 U_F 는 한 싸이클에서 서비스되는 전체 셀의 수를 round의 개수로 나눈 값으로 한 개의 round에서 평균 서비스되는 셀의 수를 나타내고 단위는 [cells/round]이다. 그림 4에서 보는 것과 같이 제안된 알고리즘이 기존의 알고리즘보다 약간의 우수한 성능 향상을 보이고 있다.

기존의 알고리즘에서는 가중치보다 적은 수의 셀이 입력된 VC에 할당된 슬롯 중에서 빈 슬롯을 처리하지 않는데 비해 제안된 셀 스케줄링 알고리즘에서는 빈 슬롯에 해당 round에서 서비스를 받지 않는 VC중에서 큐의 길이와 가중치의 차가 가장 큰 VC의 셀을 대신 할당하기 때문에 평균 서비스되는 셀의 수가 많아지게 되기 때문이다. 제안된 스케줄링 알고리즘에서는 이러한 버퍼 상태 관리 기법으로 버퍼에 셀이 차 있는 VC의 버퍼 대기 지연 시간을 줄일 수가 있으며, 그 결과 전체적인 평균 큐의 길이도 감소시키는 효과도 얻을 수가 있게 되었다.

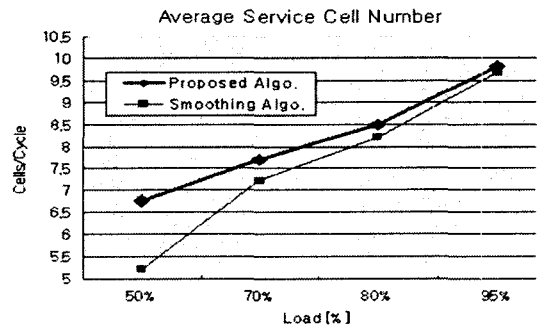


그림 5. 부하에 따른 U_F 값(실시간 · 비실시간 트래픽)

그림 5는 실시간 트래픽과 비실시간 트래픽을 함께 서비스 하는 경우에 한 개의 round에서 서비스되는 평균 셀의 수를 나타낸다. 그림 4와 비교하면 탁월한 성능 향상을 보이고 있는데, 그림 4와 다른 점은 부하가 증가함에 따라서 한 round에서 평균 서비스되는 셀 수의 증가율이 작아짐을 알 수 있다. 왜냐하면 부하가 작은 경우 빈 round와 빈 슬롯이 많아지게 되어 그만큼 비실시간 트래픽을 서비스할 확률이 높아지게 되지만 부하가 증가할수록 실시간 트래픽을 처리하는데 대부분의 슬롯을 할당하기 때문에 남은 빈 슬롯이 적어지게 되어, 그 결과 비실시간 트래픽이 서비스를 받을 기회가 줄어들게 되기 때문이다.

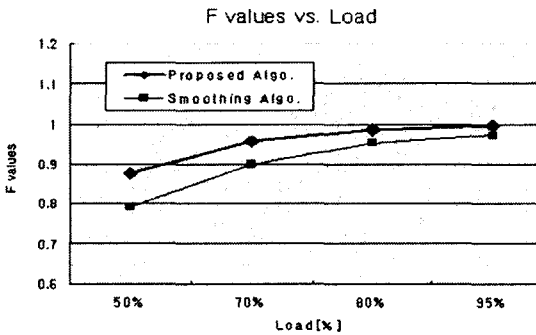


그림 6. 부하에 따른 F값의 비교(실시간·비실시간 트래픽)

그림 6은 실시간 트래픽과 비실시간 트래픽을 같이 서비스 하는 경우에 부하에 따른 파라미터 F의 값을 나타내는 그래프이다. 그림 6에서 보는 것과 같이 제안된 알고리즘이 실시간 트래픽만을 처리하는 경우보다 비실시간 트래픽도 함께 처리할 때 파라미터 F값에 대해서 더 큰 성능 향상을 보이고 있다. 왜냐하면 전체 round에 균등한 셀 배치를 위해서 빈 슬롯을 처리할 뿐만 아니라, 비실시간 트래픽을 끼워 넣음으로써 더욱 더 균등한 셀 배치가 가능해졌기 때문이다.

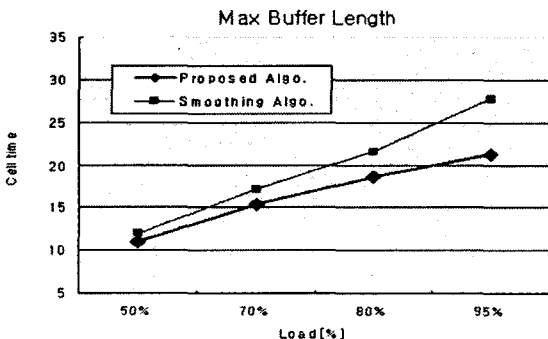


그림 7. 최대 버퍼의 길이(실시간 트래픽)

그림 7은 실시간 트래픽만을 서비스하는 경우 부하에 따른 평균 버퍼의 길이를 나타내고 있는 그래프이다. 그림 7에서 보는 것과 같이 부하가 증가함에 따라 제안된 알고리즘에서 평균 버퍼 길이의 증가율이 작아지는 향상된 성능을 보이고 있다. 기존의 알고리즘에서는 가중치를 초과하여 입력된 셀에 대한 처리를 하지 않기 때문에 전체 VC에 할당된 가중치 값의 합보다 적은 셀을 서비스하게 되는데 부하가 증가할수록 전체 VC에 할당된 가중치 값의 합과 실제로 서비스되는 셀의 수와의 차이가 증가하여 평균 버퍼의 길이가 커지게 된다. 그러나 제안된 셀 스케줄링 알고리즘에서는 부하가 증가할수록 전체 VC에 할당된 가중치 값의 합과 실제로 서비스되는 셀의 수와의 차에 대해서 버퍼 상태를 고려한 스케줄링으로 인하여 버퍼 증가율이 감소되는 결과를 얻게 되었다.

V. 결론

셀 스케줄링 알고리즘은 짧은 시간에 모든 계산을 수행해야 하므로 계산 복잡도가 작아야 한다. 이에 적합한 알고리즘이 WRR 알고리즘이다. 이러한 WRR 알고리즘을 개선한 BSM과 Smoothing Scheme과 같은 여러 알고리즘이 제안되었으나 편협한 성능 개선의 문제를 내포하고 있다. BSM과 Smoothing Scheme은 실시간 트래픽에 적합한 스케줄링 기법으로 멀티미디어 서비스를 위한 결함을 가지고 있으며 버퍼의 상태를 간과한 스케줄링으로 대역 할당에 낭비를 초래하게 된다.

본 논문에서는 이러한 문제점들을 개선한 WRR 셀 스케줄링 알고리즘을 제안하였고 기존의 알고리즘과 제안된 알고리즘의 시뮬레이션을 수행하고 성능을 평가하여 그 결과를 비교, 분석하였다. 시뮬레이션 결과에서 제안된 알고리즘이 기존의 알고리즘보다 최대 버퍼 크기는 8%~23%, 평균 서비스 셀 수는 최대 22%, 대역 할당의 공정성은 최대 9.5%까지의 향상된 성능을 확인하였다. 결과적으로 전체 트래픽에 대한 버퍼 관리와 효과적인 대역 할당으로 throughput을 높일 수 있고, 전체적으로 성능 향상을 꾀할 수 있으며, 버퍼 용량을 줄임으로써 경제적인 면에서도 이점을 가지게 되었다.

앞으로의 연구과제는 실시간 트래픽의 지연 편차를 줄이면서 비실시간 트래픽의 처리량을 향상시킬 수 있는 좀더 안정된 스케줄링 기법의 연구와 다양한 전송률을 가진 많은 VC들로 구성된 실제 망 상황과 유사한 시뮬레이션 분석이 수행되어야 하겠다. 또한 이렇게 설정된 환경에서의 스케줄링 기법의 복잡도 분석과 수학적 분석 방법이 연구되어야 할 것이다.

참 고 문 헌

[1] Yao-Tzung Wang, Tzung-Pao Lin, Kuo-Chung Gan, "An Improved Scheduling Algorithm for Weighted Round-Robin Cell Multiplexing in an ATM Switch," IEEE, pp. 1032-1037, 1998.

[2] Sung Won Lee, Dong Ho Cho, Yong Ki Park, "Improved Dynamic Weighted Cell Scheduling Algorithm based on Earliest Deadline First Scheme for Various Traffics of ATM Switch," Proceedings of the IEEE Globalcom, Vol. 3, pp. 1959-1963, Nov. 1996.

[3] Byung Kook Kim, Kang G. Shin, "Scalable Hardware Earliest-Deadline-First Scheduler for ATM Switching," Proceedings of the Real-Time Systems symposium, Dec. 1997.

[4] J. S. Cha, C. Oh, K. Kim, "A Two-phased Weighted Fair Queueing Scheme for Improving CDV and CLP in ATM networks," IEICE Transactions on Communi., V.E83-B, No. 5, pp. 1136-1139, May 2000.

[5] Chung-Shiem, Gin-Kou Ma, and Bao-Shuh P.Lin, "A Cell Scheduling Algorithm for VBR Traffic in an ATM Multiplexer," Globecom95, pp. 632-637, Sep. 1995.

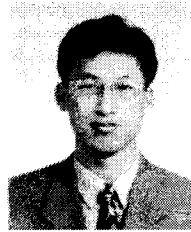
[6] 유영일, 김관웅, 조해성, 정경택, 전병실, "셀 지연과 버퍼 이용률을 개선한 WRR 셀 스케줄링 알고리즘", 통신학회 2000년도 하계학술발표회 논문집(상), pp. 272-275, July 2000.

[7] A. Arulambalam et al, "Allocating Fair Rates for Available Bit Rate Service in ATM Networks," IEEE Communications Magazine, Vol. 34, No. 11, Nov. 1996.

[8] M. Katevenis, S. Sidropoulos, and C. Courcoubetis, "Weighted round-robin cell multiplexing in a general purpose ATM switch chip," IEEE J. Sel. Areas Commun., SAC-9, pp. 1265-1279, 1991.

조 해 성(Hae-Seong Cho)

정회원



1994년 2월 : 전북대학교 전자공학과
(공학사)

1996년 2월 : 전북대학교 전자공학과
(공학석사)

2001년 2월 : 전북대학교 전자공학과
(공학박사)

2001년 9월 ~ 현재 : 건양대학교 IT학부 교수

<관심분야> : 멀티미디어 통신, 통신 프로토콜, 웹 프로그래밍