

패킷 스케줄링을 통한 실시간 인터넷 서비스의 트래픽 제어

임철수* · 이양선**

1. 개요

현재 사용되고 있는 인터넷 프로토콜(IPv4)은 주소 공간의 고갈, 강력한 보안기능의 부족 등과 같은 구조적인 문제 외에 갈수록 트래픽 비중이 커지고 있는 멀티미디어 데이터에 대한 실시간 처리기능의 부족이 큰 문제점으로 대두되고 있다. 현재의 인터넷 프로토콜은 데이터그램 패러다임에 기초하고 있으며, 패킷은 연결(connection) 개념이 없이 상호 독립적으로 라우팅 된다. 또한 서비스 품질(QoS; Quality of Service) 개념이 없어서 성능에 대한 보장이 안될 뿐만 아니라, 전송시 지연시간에 대한 한계도 제약되어 있지 않다. 즉, 기존의 IP 계층의 패킷 전달은 그 패킷의 근원이 어떠한 트래픽 소스인지에 관계없이 모두 동등하게 처리하며, 네트워크는 단지 사용자의 요구사항을 만족시키기 위해 최선형 (best effort) 서비스만을 제공할 뿐이다.

이와 같이 인터넷상에서 단순한 최선전달 서비스는 새로운 멀티미디어 어플리케이션을 수용하기에는 적합하지 않은 것으로 알려져 있는데, 이는 실시간으로 비디오나 오디오를 전송하는 것은 데이터가 지연시간이 일정한 안정된 전송율(stable rate)로 전송될 수 있도록 보장되어야 하기 때문이다. 이와 같은 관점에서 볼 때, 멀티미디어

트래픽과 기존의 IP망에서의 데이터와는 일반적으로 다음의 3가지 측면에서 그 특성이 구분된다.

- 필요전송 대역(Bandwidth) : 멀티미디어 트래픽은, 특히 영상의 경우, 빠른 전송 대역을 요구한다.
- 엄격한 시간 요구사항 : 멀티미디어 데이터는 각 모노 미디어간의 시간 관계가 매우 중요하며, 만일 필요시간 내에 데이터가 도달하지 않으면 그 데이터는 아무런 의미가 없다.
- 멀티캐스트(multicast) : 멀티미디어 데이터의 경우, 점-대-다수(point-to-many) 또는 다수-대-다수(many-to-many) 서비스가 자주 요구된다.

원하는 서비스 품질을 보장하려면, 위의 3가지 특성이 모두 만족되어야 하므로, 단순히 망의 전송능력을 늘려주는 것만 가지고는 QoS를 맞추어 줄 수 없다. 또한 구체적인 QoS 요구사항은 응용 서비스의 종류와 미디어의 성질에 따라 달라진다. 예컨대, 대화형 TV와 같은 실시간 서비스는 그것이 어느 정도의 패킷 손실율을 가져오더라도, 매우 엄격한 지연과 지연변화 요구사항을 충족시켜야 한다. 반면, 텍스트 데이터와 같은 비실시간 서비스는 엄격한 패킷 손실 요구사항을 가져야 하지만, 어느 정도의 지연은 허용한다. 참고로 ITU-T에서 정의한 QoS 요구사항 중 비디오 서비스에 대한 지연(delay) 요구사항을 정리한 것을 보면,

*. **서경대학교 전자·통신·컴퓨터공학부 교수
 **중신회원

1.5Mbps MPEG 비디오에 대한 지연시간은 5ms, 20Mbps HDTV 비디오는 0.8ms, 16kbps 압축 음성서비스는 30ms 이내일 것을 요구하고 있다.

이러한 관점에서 볼 때, 실시간 멀티미디어 서비스 제공에 필요한 네트워크 서비스 품질을 제공하기 위해서는 효율적인 패킷 스케줄링이 인터넷 IP 계층에 새로이 추가되어야 한다. 이러한 노력의 일환으로 IETF의 ISA (Integrated Service Architecture)와 DSA (Differentiated Service Architecture) 연구개발을 들 수 있다. ISA는 실시간 어플리케이션에서 발생하는 패킷의 흐름을 단위로 하여 RSVP (Resource Reservation Protocol)를 이용하여 사전에 허락 제어와 자원예약을 수행하여 QoS 보장형 서비스와 최선 전달서비스 등을 지원한다. ISA에서 QoS 보장은 결국 패킷 스케줄링 기능에 의해 실현된다. 한편, RSVP를 기반으로 한 ISA는 네트워크의 규모가 커질 때 트래픽에 비례하여 복잡성이 증대되어 확장성(Scalability) 문제가 발생하는 것으로 알려져 있다.

이에 따라 제안된 DSA 구조에서는 네트워크 상의 QoS는 PHB (Per-Hop Behavior)에 의해 결정되는데, 입력과 출력에서의 패킷의 흐름 측면에서만 규정되는 PHB는 내부적으로 다시 여러 가지 패킷 스케줄링 기능으로 실제 구현된다. 결국, DSA에서도 패킷 스케줄링 기능은 패킷 전달 서비스 품질을 향상시킬 수 있는 핵심 기능이 된다 [1].

패킷 스케줄링은 통상 라우터의 출력단에 위치하여 트래픽의 종류에 따라 패킷의 전송서비스 순서를 달리하여 결국 다양한 패킷 전달서비스를 제공하고자 하는 기능이다. 즉, QoS의 중요한 전달지연은 전적으로 네트워크에서 제공하는 서비스 실행 순서에 의존하기 때문에 패킷 스케줄링은 인터넷 트래픽 제어의 핵심요소라고 할 수 있다.

현재 인터넷 통합서비스 모델을 연구하고 있는 Intserv WG(Working Group)에서는 기존의 최선형 서비스 (BE: best-effort service) 외에, 보장형 서비스 (GS : Guaranteed Service)와 부하제어형 서비스 (CLS: controlled load service)를 정의하고 있다 [2,3]. 이와 같이 인터넷에서 종합서비스를 제공하기 위해서는 인터넷에서 이를 지원하기 위한 기술이 필요하다. 특히 실시간 서비스를 보장하기 위해서는 자원예약 프로토콜과 효율적인 트래픽 제어부가 구현되어야 하는데, 트래픽 제어부는 패킷 스케줄러, 분류자, 수락 제어의 3가지 요소들에 의해서 구현된다. 그림 1은 이러한 구성 요소들을 라우터에 적용한 모델이다 [2].

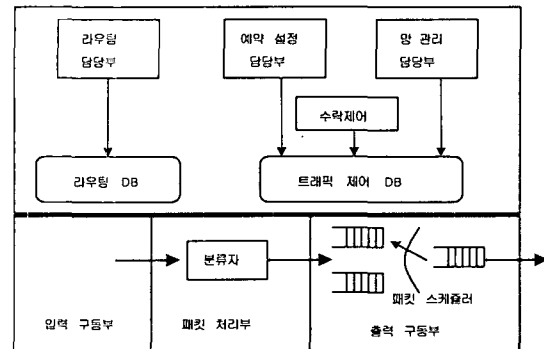


그림 1. 라우터에서의 종합서비스 구현 모델

그러나, 오늘날 대부분의 인터넷 라우터들은 FCFS (First-Come, First-Served) 방식으로 동작한다. 즉, 입력라인으로 도달하는 패킷들은 라우팅 프로세스에 의하여 검사된 후 출력 라인으로 큐잉되는데, 도착 순서대로 전송되게 된다. 이와 같은 FCFS 방식은 구현이 쉬운 장점이 있는 반면, 지연 시간이 네트워크 부하(load)에 따라 달라질 뿐만 아니라, 모든 흐름들에게 요구에 따른 공평한 서비스 품질을 제공하므로 다양한 QoS를 요구하는 실시간 서비스를 제공하기에는 적합하지

못하다. 따라서, 이와 같은 문제점을 해결하기 위한 많은 효율적인 알고리즘들이 제안되어 왔다 [4-9].

이에 따라, 본 논문의 제2장에서는 현행의 인터넷에 널리 적용중인 FCFS 방식의 문제점을 지적하고, 여러 스케줄링 알고리즘들을 실시간 서비스 관점에서 적합성 및 각각의 장·단점을 비교하였다. 또한 제 3장에서는 인터넷에서 실시간 어플리케이션을 지원하는데 요구되는 지연시간의 제약성 및 공평성 서비스 분석을 통해 어떤 스케줄링 알고리즘이 인터넷에서 실시간 멀티미디어 어플리케이션을 서비스하는 것이 적합한지를 보여주고자 한다.

2. 스케줄링 알고리즘

큐에서 패킷을 우선권에 따라 서비스하는 우선순위 제어방법은 간단하기는 하지만, QoS를 보장해 줄 수 없기 때문에 이보다 다소 복잡하기는 하지만 사전에 지정된 대역을 보장해주는 방식이 있다. 일반적으로 이러한 대역보장형 스케줄링 방식은 작업보존(work conserving) 방식과 비 작업보존 방식으로 구분된다. 이때 스케줄러가 작업보존 방식이라는 것은 패킷이 시스템에 버퍼링되어 있으면 서버가 결코 휴지(idle)상태에 있지 않음을 나타내며, 비작업보존 방식이란 전송해야 할 패킷이 있더라도 서버가 휴지상태가 될 수 있는 스케줄링 방식을 말한다. 따라서, 여기에서는 현재 많은 연구가 되고 있는 작업보존 방식에 속하는 제반 스케줄링 알고리즘들의 기능 및 장·단점을 비교해 보고자 한다.

2.1 Fair Queueing

이는 가장 간단한 스케줄링 방식으로서, 각각

의 데이터 소스로부터의 패킷들에 대하여 별도의 큐를 유지하고 각 흐름(또는 큐)에 대해 라운드-로빈 방식으로 서비스를 제공한다. 이는 FCFS 보다 패킷 처리율, 평균 왕복시간 등의 측면에서 성능이 개선된 방식이다. 그러나, 모든 흐름에 대한 대역폭이 같고 패킷의 길이가 모두 동일할 때에만 적합하며, 각 흐름에 할당된 대역폭이 서로 다르고 패킷의 길이가 다를 경우에는 적합하지 않다.

2.2 GPS (Generalized Processor Sharing 또는 Fluid-flow FQ)

각 흐름마다 다른 비율로 대역폭을 할당받을 수 있도록 하는 방식으로, 각 흐름에 대한 서비스가 무한히 작은 단위로 이루어질 수 있다는 가정하에 최악의 상황에서도 할당된 대역폭을 보장할 수 있는 이상적인 방법이다. 그러나, 실제로 모든 전송은 패킷단위로 하고, 하나의 서버로 모든 흐름을 동시에 서비스할 수 없기 때문에 구현이 불가능하다.

2.3 PGPS (Packet Generalized Processor Sharing 또는 WFQ)

이는 GPS 방식에 가장 근사한 알고리즘으로서, GPS를 시뮬레이션 하여 각 흐름에 속한 패킷 중에서 GPS 방식으로 서비스될 때 서비스가 종료되는 순서대로 서비스 되도록 선택하여 처리하는 방식이다. 그러나, PGPS는 가상 시간을 계산하기 위해 GPS를 시뮬레이션 해야하기 때문에 고속망에서는 구현이 어렵다. PGPS의 지연한계(delay bound)는 GPS와 비교할 때, 하나의 패킷 전송시간 이내 정도로 거의 같은 수준의 지연시간을 가지고 있지만 [4], 두 시스템에서 제공하는 서비스 차이는 크다고 할 수 있다.

다음 [그림 2]는 링크를 공유하는 11개의 세션들을 나타내고 있다. FCFS, PGPS(WFQ), WF²Q 방식에 따른 서비스 순서를 간단히 설명하기 위해 모든 패킷들이 크기 1로써 같고, 링크 속도가 1이라고 가정해보자. 또한 세션 1에 보장된 전송율은 0.5이고 다른 10개 세션 각각에 대하여 보장된 전송율은 0.05라고 가정하자.

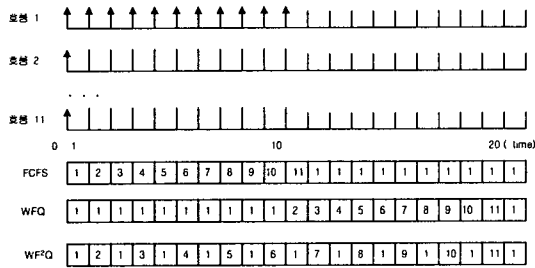


그림 2. FCFS, WFQ, WF²Q 서비스 순서

2.4 WF²Q (Worst-case Fair Weighted Fair Queueing)

GPS를 더욱 근사화한 방식으로 WFQ가 GPS 시스템에서 패킷 종료시간만을 사용하는 반면에, 보다 정확하게 GPS방식을 에뮬레이션하기 위해 시작시간과 종료시간 두 가지를 사용하여 서비스 순서를 결정한다. 이에 따른 서비스 순서는 위 그림 2 마지막 부분에 나타나 있다. 이는 GPS와 거의 유사한 성능을 가지고 최소한의 지연시간을 보장받을 수 있을 뿐만 아니라 공평성을 제공할 수 있는 이점이 있는 반면, GPS를 시뮬레이션하여 가상 시작시간 및 가상 종료시간을 계산해야 하므로 고속망에서는 구현이 더욱 더 어렵다.

2.5 SCFQ (Self-Clocked Fair Queueing)

이는 가상시간에 대한 계산 오버헤드를 줄이기 위해, 시스템 자체에서 정의한 가상시간을 이용하

는 방법으로서 현재 서비스중인 패킷에 타임스탬프(timestamp) 라는 가상시간을 붙이는 방법이다. 도착하는 패킷은 PGPS에서와 같이 가상 종료시간을 계산하여 붙이고, 이 값이 적은 패킷을 우선적으로 서비스하는 방식이다. 이 방식은 가상시간에 대한 계산 오버헤드가 적어서 구현이 쉽고 각 흐름의 최대 지연시간이 보장되지만, 계산량이 흐름의 연결 수에 따라 커지는 문제가 있다.

2.6 Virtual Clock

시분할 다중화 (TDM: Time Division Multiplexing) 방식을 시뮬레이션하여 각 흐름을 서비스하는 방식이다. 각 흐름에 속한 패킷은 TDM 방식으로 서비스하는 경우의 가상 종료시간을 계산하여 그 값이 가장 작은 패킷부터 서비스한다. 모든 연결에 대하여 가상 종료시간을 계산해야 하기 때문에 대역폭을 초과해서 부하가 증가하게 되면 보장받은 대역을 사용하지 못하는 문제가 발생한다 [10].

3. 스케줄링 알고리즘에 대한 성능 분석

트래픽 스케줄링 알고리즘이 인터넷에서 효율적인 트래픽 제어를 위해 실제로 유용하게 사용되기 위해서는 일차적으로 다음과 같은 특성들을 고려하여야 한다.

- 낮은 종단간 지연(Low end-to-end delays) : 개별 세션에 대하여 종단간 지연시간을 보장할 수 있어야 한다.
- 공평성(Fairness) : 링크를 공유하는 모든 세션들에 대하여 가용 대역폭을 공평하게 나누어 서비스해야 한다.
- 구현 복잡도(Implementation Complexity) :

스케줄링 알고리즘은 신속한 처리를 위해 고속망에서 하드웨어적으로 구현될 필요가 있으며, 알고리즘에 대한 계산시간 복잡도가 현재 사용중인 연결 수(V)와 무관한 것이 바람직하다.

이외에도 스케줄링 알고리즘에는 다음과 같은 특성들이 반영되어야 한다.

- 독립된 흐름(Isolation of flows) : 알고리즘은 특정 세션이 잘못 동작하고 있는 다른 세션들로부터 영향을 받지 않고, 정해진 QoS를 유지할 수 있어야 한다.
- 확장성(Scalability) : 알고리즘은 다양한 링크 속도는 물론, 연결 개수가 많은 스위치에서도 잘 동작되어야 한다.

이러한 특성 중에서 본 단원에서는 알고리즘들간에 차이가 큰 지연시간과 공평성의 관점에서

알고리즘들을 분석해 보면 다음 표1과 같이 요약된다 [8]. 이 표는 작업보전방식에서의 여러 알고리즘들에 대한 지연시간(latency), 공평성, 그리고 구현 복잡도를 나타내고 있다. 여기에서, r 은 링크 속도 또는 서버의 전송율을 나타내고, L_{max} 는 모든 세션중 패킷의 최대길이를 나타내며, L_i 는 세션 i 에서 최대 패킷 크기를 각각 나타낸다. 또한, V 는 서버(또는 연결)의 개수를, C_i 는 PGPS 서버에서 하나의 세션이 제공받을 수 있는 최대 서비스 시간을, F 는 프레임 크기를, ϕ_i 는 세션 i 에 할당된 트래픽량 각각을 나타낸다.

이 표에 따르면, PGPS 스케줄러는 지연시간과 공평성 측면에서는 가장 좋은 성능을 가진 것으로 나타나 있다. 그러나, 구현상의 복잡도 측면에서는 가장 높은 값을 가지고 있다. 반면, VirtualClock 알고리즘은 PGPS와 동일한 지연시간 한계를 가지지만, 공평성이 매우 낮다.

표 1. 스케줄링 알고리즘들의 성능 분석

알고리즘	지연시간(Latency)	공평성(Fairness)	구현복잡도(Complexity)
GPS	0	0	-
PGPS	$\frac{L_{max}}{r} + \frac{L_i}{\rho_i}$	$\max(\max(C_j + \frac{L_{max}}{\rho_j} + \frac{L_j}{\rho_j}, C_i + \frac{L_{max}}{\rho_i} + \frac{L_i}{\rho_i}),$ $where$ $C_i = \min((V-1)\frac{L_{max}}{\rho_i}, \max_{1 \leq n \leq V}(\frac{L_n}{\rho_n}))$	$O(V)$
SCFQ	$\frac{L_{max}}{r} * (V-1) + \frac{L_i}{\rho_i}$	$\frac{L_i}{\rho_i} + \frac{L_j}{\rho_j}$	$O(\log V)$
Virtual Clock	$\frac{L_{max}}{r} + \frac{L_i}{\rho_i}$	∞	$O(\log V)$
Frame-Based Fair Queueing	$\frac{L_{max}}{r} + \frac{L_i}{\rho_i}$	$\max(\frac{2F - \phi_i}{r} + \frac{L_i}{\rho_i}, \frac{2F - \phi_j}{r} + \frac{L_j}{\rho_j},$ $\frac{L_{max}}{\rho_j} + \frac{L_i}{\rho_i}, \frac{L_{max}}{\rho_i} + \frac{L_j}{\rho_j})$	$O(\log V)$

3.1 처리 지연 한계(Delay bound)

작업보존방식에 속하는 대부분의 스케줄러는 세션 트래픽에 대한 버스티 정도가 제한되어 있을 때 정해진 지연시간을 보장할 수 있으므로 종단간의 정해진 실시간 제약사항을 어느 정도 만족시켜 줄 수 있다. 따라서, 여기에서는 현재 널리 적용되고 있는 SCFQ 스케줄러에 대한 지연 한계값을 분석적으로 제시함으로써, 제반 알고리즘들도 이와 비슷한 한정적인 지연시간을 가짐을 보이고자 한다.

여기서는 SCFQ 스케줄러에 따른 지연시간에 대한 상한이 $(V-1)\frac{L_{max}}{r} + \frac{L_i}{\rho_i}$ 임을 증명하기 위해서 이러한 경우가 발생하는 실제 경우를 예로서 제시하고자 한다. L_i 를 세션 i 에서의 최대 패킷 크기라 하고, ρ_i 를 세션 i 에 할당된 대역폭 비율, 그리고 나머지 $(V-1)$ 개의 연결에는 $\frac{(1-\rho_i)}{(V-1)}$ 비율만큼 대역폭이 할당되었다고 가정하자. 또한 세션 i 를 제외한 모든 연결들이 0 시점에 L_{max} 크기의 패킷만큼 백-로깅(backlogging) 되어 있다고 가정하자. 이에 따라 나머지 연결들은 TS_{cur} 만큼의 동일한 가상 종료시간을 갖게 된다. 이러한 연결중의 어느 하나에서 첫 번째 패킷이 서비스되고 나면, 크기 L_i 인 패킷이 세션 i 로부터 도달한다. 그러면,

$$F_i^1 = TS_{cur} + \frac{L_i}{\rho_i} \text{ 이 된다.}$$

이때, 세션 i 의 백로깅 기간에 속한 이 패킷과 나머지 모든 패킷들은 가상 종료시간이 TS_{cur} 인 패킷들에 의해 지연될 것이다.

이제 $j \neq i$ 인 각각의 세션이 $S = \frac{L_i}{\rho_i} \rho_j$ 크기 만큼의 패킷을 전송한다고 가정하자. 이 그룹에 속한 마지막 패킷의 가상 종료시간은 각각의 세션 j 에 대하여, $TS_{cur} + \frac{L_i}{\rho_i}$ 가 될 것이다. 따라서, 최악의 경우에, 이러한 모든 패킷들이 서비스된 후에 세션 i 로부터의 패킷이 서비스를 받게 될 것이다.

따라서, 세션 i 에서의 첫 번째 패킷은 다음 식에 주어진 시간에 종료하게 된다.

$$\begin{aligned} D_i &= (V-1)\frac{L_{max}}{r} + \frac{1}{r} \sum_{j=1, j \neq i}^V \frac{L_i}{\rho_i} \rho_j + \frac{L_i}{r} \\ &= (V-1)\frac{L_{max}}{r} + \frac{L_i}{\rho_i} \frac{r-\rho_i}{r} + \frac{L_i}{r} \\ &= (V-1)\frac{L_{max}}{r} + \frac{L_i}{\rho_i} \end{aligned}$$

이에 따라 SCFQ 알고리즘에서의 지연시간 한계는 $(V-1)\frac{L_{max}}{r} + \frac{L_i}{\rho_i}$ 로서 기존에 제시된 방식들 중에서 가장 적다고 할 수 있다. 이를 그림으로 나타내어 설명하면 다음과 같다.

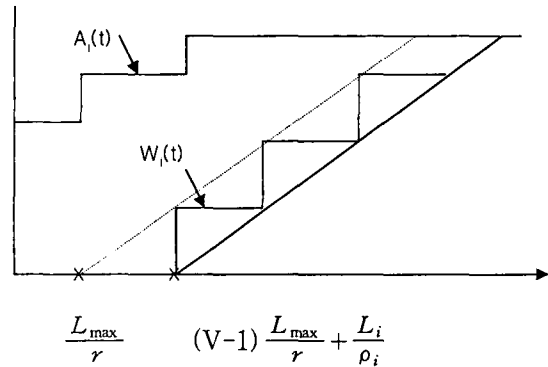


그림 3. 세션 i 에서의 서비스 제약 시간

$A_i(\tau, t)$ 를 $(\tau, t]$ 간격 동안 세션 i 로부터의 도착 트래픽을 나타내고, $W_i(\tau, t)$ 는 $(\tau, t]$ 간격 동안 세션 i 가 제공한 서비스 량이라고 하면, 양자는 모든 시간에 대한 연속함수로 나타낼 수 있다. 이에 따라 t 시점에 세션 i 에서 큐잉된 트래픽 양은 다음과 같이 나타낼 수 있으며, $Q_i(t) > 0$ 인 경우에 트래픽이 백로깅(back-logging) 되어있다고 할 수 있다.

$$Q_i(t) = A_i(0, t) - W_i(0, t)$$

위의 [그림 3]은 하나의 세션에 대한 서비스 제약시간을 2개의 경계 값(상한과 하한) 을 보여주

고 있으며, 하한의 경계치는 세션이 전송 중 (session-busy)일 때 어떤 시점에서도 적용되며, 상한 경계치는 패킷이 시스템을 떠나는 시점에만 유효하다.

3.2. 공평성(fairness)

스케줄링 알고리즘의 공평성을 추정하는데 있어서, 모든 경우에 적용될 수 있는 보편적인 방법은 없다. 그러나, 일반적으로는 시스템에서 연결들 각각에 대한 예약율에 따라 서비스를 제공하고, 이전에 서비스 받은 것에 대하여 향후 아무런 불이익이 없는 경우에 공평한 스케줄러라고 말할 수 있다.

어떤 τ 시점에서 i 와 j 라는 2개의 연결이 무한 양의 대역폭을 요구한다고 가정하면, 두 개의 연결은 τ 시점 이후에는 계속적으로 백로킹 될 것이다. 이때 τ 시점 이후 (t_1, t_2) 동안에 i 와 j 라는 2개의 연결에 제공된 평균화된 서비스 (normalized service)의 차이가 한정되어 있으면, 스케줄러가 공평하다고 할 수 있다. 즉,

$$\left| \frac{\overline{W}_i(t_1, t_2)}{\rho_i} - \frac{\overline{W}_j(t_1, t_2)}{\rho_j} \right| \leq FR$$

여기에서 $FR < \infty$ 값은 알고리즘의 공평성에 대한 측정치가 되며, 이는 곧 인터넷상에서의 다양한 사용자들에게 공평한 서비스를 제공할 수 있는지를 가늠하는 척도가 된다.

패킷 스케줄링 알고리즘에 대한 불공평성 (unfairness)의 대표적인 예는 [그림 4]와 같이 VirtualClock 알고리즘에 잘 나타나있다. 두 개의 연결이 하나의 출력 링크를 공유하고 있고, 이들에게는 동일한 비율로 대역폭이 할당되어 있다고 가정하자. 또한 각 패킷의 크기는 1이고 서버에서의 전송율도 1이라고 가정하자. 0~1000 시점 동

안에 세션 1 만이 액티브(active)되어 있어서 1000 개의 패킷을 보낸다고 하자. 세션 2는 1000시점에 액티브 되어서, 1000 시점 이후에는 2개의 세션이 패킷을 보낸다고 하자.

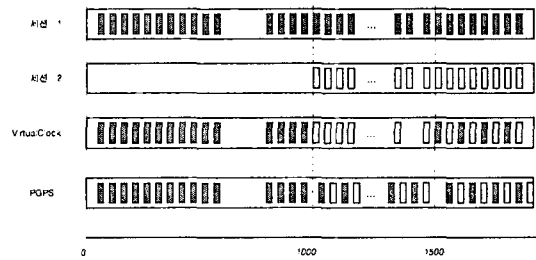


그림 4. VirtualClock 알고리즘의 불공평성

서버에서의 스케줄러가 VirtualClock 알고리즘 방식으로 구현되어 있다면, 서버는 작업보존 방식이므로 $t=1000$ 시점까지는 세션 1로부터의 1000 개의 패킷을 모두 서비스하게 될 것이다. 그러나, $t=1000$ 시점 다음에 세션 1을 통해 들어오는 패킷 들은 1000시점까지 받은 평균 서비스 시간을 감안 하여 타임스탬프 값으로 2000을 배정 받게 된다. 따라서, 두 세션에 대한 타임스탬프 값이 같아지는 1500 시점까지 세션 1은 전혀 서비스를 받지 못하게 된다. 만약 데이터 소스에서의 최대 버스티 (bursty) 트래픽량이 제한되어 있지 않다면, 백로킹에 따라 서비스를 받지 못하는 구간의 크기가 ∞ 로 커질 수 있다. 반면에, PGPS 서버는 이전에 세션 1에서 서비스 받은 대역폭에 관계없이, $t=1000$ 시점 이후에 2개의 세션에 대하여 동등한 서비스를 제공한다.

4. 결론

본 논문에서는 현재 인터넷상에서 트래픽 비중이 급증하고 있는 멀티미디어 데이터에 대하여

실시간 서비스를 제공하기 위해 효율적인 패킷 스케줄링 알고리즘을 통해 트래픽을 제어하는 방안을 검토하였다. 이와 같이 인터넷상에서 패킷 전달 서비스 품질 향상에 핵심적인 패킷 스케줄링 방식들은 QoS 보장 및 처리 효율성, 그리고 구현 측면에서 각각의 장·단점을 가지고 있다. 이에 따라, 본 논문에서는 다양한 패킷 스케줄링 알고리즘들의 중요성과 기능을 비교하여 설명하고, 이러한 알고리즘이 인터넷에서 실시간 멀티미디어 서비스를 효과적으로 제공할 수 있는지 여부를 지연시간과 서비스 공정성 측면에서 분석하였다.

작업보전 방식의 GPS는 가장 이상적인 스케줄링 방식이나 실제 구현이 불가능하므로, 이를 근사화한 PGPS(WFQ) 방식은 최소한의 지연 변이에 대한 보장 및 종단간의 QoS를 보장한다. 또한 분석결과에 따르면, VirtualClock 알고리즘을 제외한 다른 알고리즘들은 공평한 서비스를 제공하는 것으로 간주된다. SCFQ 방식은 휴지(idle) 기간 후에 패킷이 백로그되어 있을 때는 세션에 대한 서비스를 지연시키는 반면, PGPS에서는 새로이 백로그된 세션을 서비스하기 위해 이미 서비스를 받은 세션에 대해서는 불이익을 부여한다. 따라서, SCFQ 알고리즘은 패킷 처리 스케줄링 방식 중에서, 어떤 경우에는 PGPS 방식보다도, 가장 공정성이 좋은 반면에, 지연시간은 PGPS 방식보다 더 높다고 할 수 있다.

참 고 문 헌

- [1] S. Blake et al, "An Architecture for Differentiated Services", Internet-Draft Diffserv Working Group, Oct. 1998.
- [2] R. Braden et al., "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994.
- [3] S. Shenker, C. Partridge, and R. Guerin, "Specification of Guaranteed Quality of Service", Internet RFC 2212, Sep 1997.
- [4] A. Parekh and R. Gallager, "A Generalized processor Sharing approach to flow control - the single node case", ACM/IEEE Transactions on Networking, 1(3): 344-357, June 1993.
- [5] S. Jamaloddin Golestani, "A Self-Clocked Fair Queueing Scheme for Broadband Applications", Proc. of IEEE INFOCOM'94, pp.636-646, June 1994
- [6] Fabio M. Chiussi and Andrea Francini, "Implementing Fair Queueing in ATM Switches", Proc. of GLOBECOM'97, November 1997.
- [7] J.C.R. Bennett, H. Zhang, "WF²Q+: Worst-case fair weighted fair queueing", Proc. of IEEE INFOCOM'96, pp.120-128, March 1996
- [8] D. Stiliadis, and A. Varma, "Latency-Rate Servers : A General Model for Analysis of Traffic Scheduling Algorithms", UCSC-CRL-95-38, 1995년
- [9] D. Stiliadis, and A. Varma, "Frame-based Fair Queueing : A New Traffic Scheduling Algorithm for Packet-Switched Networks", UCSC-CRL-95-39, 1995년
- [10] 한국전자통신연구원, ATM 상의 인터넷 서비스 기술개론, 진한도서, 1999년
- [11] Chrisrian Huitema, Routing in the Internet - 2nd ed, Prentice Hall PTR, 2000년.

[1] S. Blake et al, "An Architecture for Differ-



임 철 수

- 1985년 서울대학교 계산통계학과 (학사)
- 1988년 Indiana University(미) 전산과학과 (석사)
- 1994년 서강대학교 전자계산학과 (박사)
- 1985~1996년 (주)데이콤, (주)아시아나항공, (주)신세기통신 근무
- 1997년 3월~현재 서경대학교 전자·통신·컴퓨터공학부 교수
- 관심분야 : 멀티미디어통신, 차세대인터넷 응용기술



이 양 선

- 1985년 동국대학교 컴퓨터공학과 (공학사)
- 1987년 동국대학교 대학원 컴퓨터공학과 (공학석사)
- 1993년 동국대학교 대학원 컴퓨터공학과 (공학박사)
- 1994년 3월~현재 서경대학교전자·통신·컴퓨터공학부 교수
- 2000년 2월~현재 멀티미디어학회 이사, 학회지편집위원
- 관심분야 : 멀티미디어시스템, 분산객체시스템, 프로그래밍언어 및 시스템소프트웨어