Journal of Korea Multimedia Society Vol. 17, No. 12, December 2014(pp. 1453-1460)
http://dx.doi.org/10.9717/kmms.2014.17.12.1453

차별화된 패킷 손실률을 보장하는 가중치 기반 공정 큐잉 스케줄러

김 태 준†

A Weighted Fair Queuing Scheduler Guaranteeing Differentiated Packet Loss Rates

Tae Joon Kim†

ABSTRACT

WFQ (Weighted Fair Queuing) provides not only fairness among traffic flows in using bandwidth but also guarantees the Quality of Service (QoS) that individual flow requires, which is why it has been applied to the resource reservation protocol (RSVP)-capable router. The RSVP allocates an enough resource to satisfy both the rate and end-to-end delay requirements of the flow in the condition of no packet loss, and the WFQ scheduler guarantees those QoS requirements with the allocated resource. In practice, however, most QoS-guaranteed services allow a degree of packet loss, especially from 0.1% to 3% for Voice over IP. This paper discovers that the packet loss rate of each traffic flow is determined by only its time-stamp adjustment value, and then enhances the WFQ to provide a differentiated packet loss guarantee under general traffic conditions in terms of both traffic characteristics and QoS requirements. The performance evaluation showed that the proposed WFQ could increase the utilization of bandwidth by 8~11%.

Key words: WFQ, Packet Scheduling, Packet Loss Differentiation, Packet Loss Guarantee, Quality of Service

1. 서 론

IETF(Internet Engineer Task Force)에서 차등 서비스(DiffServ) 모델과 종합 서비스(IntServ) 모델을 제시하여 서비스 품질 지원뿐만 아니라 품질 보장까지 요구하는 인터넷 전화, 인터넷 영상회의와 같은 실시간 멀티미디어 통신 서비스 수용의 토대를 마련하였다[1]. 그 동안 대역폭과 종단간 지연 위주의 서비스 품질이 주로 다루어져 왔지만 최근 들어 패킷 손실률의 차별화[2]와 차별화된 패킷 손실률의 보장[3]에 대한 연구도 진행되고 있다.

이상적 패킷 스케줄링 방식인 GPS[4]는 WFQ (Weighted Fair Queuing)[5]에 의해 실현되었다. WFQ는 흐름 간 공평한 자원 사용을 지원할 뿐만 아니라 흐름들의 상호 간섭을 차단하여 개별 흐름의 품질특성을 지원하고 흐름별 주문화된 품질을 보장할 수 있어 IntServ 모델의 RSVP 라우터에 적용되었고[1], WFQ 기반 RSVP-라우터 경로의 부하 밸런싱으로 네트워크의 트래픽 수용 능력을 높일 수 있음을 보였다[6]. WFQ는 요구되는 대역폭과 지연 한계를 완벽히 준수하여 패킷 손실이 전혀 발생하지 않는 일종의 무손실 패킷 스케줄링 알고리즘이다.

※ Corresponding Author : Tae Joon Kim, Address: (331-717) 1223-24 Cheonan Daero, Seobuk-gu, Cheonan-si, ChungNam, Korea, TEL : +82-10-5426-0362, FAX : +82-41-556-6447, E-mail : tjkim@kongju.ac.kr

Receipt date : July 31, 2014, Revision date : Oct. 15, 2014
Approval date : Nov. 2, 2014
† Div. of Information & Communication Eng., College of Eng., Kongju National University

품질 보장형 서비스라 하더라도 어느 정도의 패킷 손실, 특히 인터넷 전화의 경우 0.1~3%의 패킷 손실률이 허용되며, 구체적인 손실률은 사용된 코덱과 요구 품질에 의존한다[7]. 최근 들어 무손실 WFQ에 패킷 손실을 수용하는 연구가 시도되고 있다. 다수의 차등 서비스(DiffServ) 흐름을 수용하며 WFQ 스케줄러로 실현되는 MPLS 터널에서의 패킷 손실률을 추정하고, 추정된 손실률이 요구 패킷 손실률을 초과할 경우 적응 가중치 기법으로 링크 대역폭을 동적으로 재할당하여 패킷 손실률의 보장을 시도하였다[8]. 하지만 다수의 흐름 집합에 대한 것으로 개별 흐름의 패킷 손실률 보장에는 적용할 수 없으며, 가용 대역폭이 없을 경우 손실률 보장이 불가능한 단점이 있다. 한편 [9]에서 WFQ에 패킷 손실을 허용하는 대신 트래픽 흐름을 더 많이 수용하여 자원 이용도를 크게 향상시켰고, [10]에서 이를 발전시켜 트래픽 흐름별 부하 특성과 요구 서비스 품질이 모두 동일한 제한된 환경에서 WFQ에서 개별 흐름의 패킷 손실률의 차별화와 이의 보장에 대해 연구하였다. 본 논문에서는 [9]와 [10]의 연구결과를 바탕으로 트래픽 특성뿐만 아니라 요구 서비스 품질까지 다양한 일반적인 트래픽 환경에서 차별적 패킷 손실률을 보장하는 손실 WFQ 스케줄러를 제안하고, 손실률 보장을 위한 최적 타임스탬프 조정값을 보다 빨리 구할 수 있는 알고리즘을 연구한다.

2. 관련 연구

유체흐름 기반의 이상적인 스케줄링 방식인 GPS를 구현한 것으로 패킷 손실이 전혀 없는 무손실 WFQ[5]와 흐름의 요구 패킷 손실률에 맞는 패킷 손실을 허용하여 자원 이용도를 높이는 손실 WFQ를 살펴본다.

2.1 무손실 WFQ

WFQ에서 트래픽 흐름이 스케줄러 출력링크의 대역폭을 점유하는 비율로 정의되는 가중치가 Φ_i인 임의의 흐름 i에 할당되는 대역폭 R_i는 다음과 같이 계산된다.

$$R_i = \Phi_i C \tag{1}$$

(1)에서 C는 출력링크 대역폭을 의미한다. WFQ는 처리할 트래픽이 있는 구간(이하, 바쁜 구간) 동안 다음 식에 의해 증가되는 서버시간 $v_S(t)$을 도입하여 사용한다.

$$v_S(t) = \frac{1}{\sum_{k \in B(t)} \phi_k(t)} t \tag{2}$$

(2)에서 $B(t)$는 시간 t 시점에 이미 스케줄러에 도착한 흐름, 즉 백로그(backlog) 흐름의 집합을 의미한다. 여기서 흐름이 도착한다는 것은 RSVP에 의해 수락된 흐름의 첫 번째 패킷이 스케줄러에 입력되는 것을 말한다. 서버시간은 스케줄러가 처리한 트래픽량 측면에서의 시간을 의미하며 처리할 트래픽이 없는 구간(이하, 휴지 구간) 동안 0의 값을 갖는다.

WFQ는 패킷 단위로 스케줄링 작업을 수행하므로 패킷의 가상 전송시작 시각과 가상 전송종료 시각 개념을 도입하여 사용한다. 이들 시각은 대응되는 GPS에서의 전송 시작시각과 전송 종료시각과 각각 일치한다. 임의의 흐름 i의 k번째 패킷을 P_i^k, 이의 가상 전송 시작시각과 가상 전송 종료시각을 각각 S_i^k와 F_i^k, 그리고 이의 크기를 L_i^k로 표현한다. [4]에 따르면 F_i^k는 (3)에 의해 계산되고, 이를 P_i^k의 타임스탬프(timestamp)라 부르며, 스케줄러에서 바로 그 패킷의 전송 우선순위가 된다.

$$F_i^k = \max(v_s(t), F_i^{k-1}) + L_i^k / R_i \text{ 및 } F_i^0 = 0. \tag{3}$$

WFQ에서 임의 흐름의 예약속도 R과 패킷의 최대 지연 $d^{\max}$와의 관계는 다음과 같이 주어진다[4]. 참고로 예약속도인 R은 RSVP 절차에 의해 결정된다.

$$d^{\max} = \frac{L}{R} + \frac{L^{\max}}{C} \tag{4}$$

여기서 L은 흐름의 최대 패킷 크기, $L^{\max}$는 스케줄러가 취급하는 최대 패킷 크기이다. 일반적으로 $L/R >> L^{\max}/C$이므로 (4)로부터 최대지연과 예약속도는 반비례함을 알 수 있다. 어떤 흐름에 보장되는 최대지연과 속도는 바로 $d^{\max}$와 R이 되며, R 값의 조정으로 원하는 최대지연 값을 얻을 수 있다. 여기서 최대지연은 바로 해당 스케줄러에서 허용되는 흐름의 지연한계, 즉 지연규격이 된다.

2.2 손실 WFQ

무손실 WFQ는 최대로 수용할 수 있는 흐름들의 가중치 합, 즉 허용 누적 가중치 ψ^L을 1 이하로 통제

하지만 손실 WFQ는 1을 초과하도록 하여 패킷 손실을 허용하는 대신 더 많은 흐름을 수용하는, 즉 과잉 흐름 수락을 시도한다[9,10].

손실 WFQ는 스케줄러가 휴지상태일 때 0의 값을 갖고 바쁜 상태일 때 실시간 속도로 진행하는 서버 실시간 타임머 $t_R()$을 사용한다. [4]에 따르면 WFQ에서 임의의 흐름 i의 k번째 패킷 P_i^k의 실제 전송 종료시각인 $t_R(P_i^k$전송종료)은 그 패킷의 타임스탬프 값 F_i^k에 패킷 전송 소요시간의 최대값인 $L^{\max}/C$를 더한 값 보다 늦을 수 없다. 즉 $t_R(P_i^k$전송종료)$\leq(F_i^k+L^{\max}/C)$. 이로부터 패킷의 전송 소요시간이 $L^{\max}/C$보다 길 수 없으므로 P_i^k가 지연규격을 위반하지 않으려면 $t_R(P_i^k$전송시작 직전)$\leq F_i^k$의 조건이 만족되어야 함을 알 수 있다. 따라서 손실 WFQ는 P_i^k의 전송시작 직전에 $t_R(P_i^k$전송시작 직전)$\leq F_i^k$의 조건이 충족되는지 조사하고, 충족되지 않으면 그 패킷을 폐기한다.

[9]는 트래픽 흐름별로 패킷 손실률을 차별화 할 수 없어 가장 엄격한 패킷 손실률을 위반하지 않는 범위내에서 흐름을 과잉 수락한다. 이 결과 느슨한 패킷 손실률을 요구하는 흐름에 대해서도 쓸데없이 엄격한 패킷 손실률을 보장하느라 자원을 낭비하게 된다. 이러한 문제점을 해결하기 위해 [10]에서 패킷의 타임스탬프 값을 강제로 조정하고, 조정되는 값의 정도를 제어하여 트래픽 흐름별 패킷 손실률을 차별화 하였다. 하지만 트래픽 흐름별 부하 특성과 요구 서비스 품질이 모두 동일한 특정 환경을 대상으로 하였으며, 손실률 보장을 위한 최적 타임스탬프 조정값을 구하는 시간이 많이 소요되는 문제점이 있다.

3. 제안 방법

먼저 타임스탬프 조정값과 트래픽 흐름의 패킷 손실률의 상관관계를 분석한 후 일반적인 트래픽 환경에서 차별화된 패킷 손실률을 보장할 수 있는 손실 WFQ를 제안하고, 최적 타임스탬프 조정값을 보다 효과적으로 구할 수 있는 알고리즘을 기술한다. 본 논문에서 제안하는 손실 WFQ를 GLWFQ(General Lossy WFQ)라 부른다.

3.1 패킷 손실률의 특성 분석

트래픽 흐름 별 패킷 손실 정도와 흐름의 타임스탬프 조정값 ΔT와의 관계를 살펴본다. GLWFQ 스케줄러가 다음 패킷을 출력링크로 전송하려는 시점인 t_p에 인가된 트래픽이 출력링크 용량을 초과하는 과부하가 발생하고, 이로 인해 일부 패킷이 지연규격을 위반하게 된다고 가정하자. t_p 시점에서 임의의 흐름 i의 HoL(Head-of-Line) 패킷 P_i^{HoL}의 우선순위 값 U_i^{HoL}은 다음과 같이 타임스탬프와 타임스탬프 조정값의 합으로 계산된다.

$$U_i^{HoL}=F_i^{HoL}+\Delta T_i \tag{5}$$

GLWFQ 스케줄러는 우선순위 값이 가장 작은, 즉 $Min_{i\in B(t)}(U_i^{HoL})$인 패킷 P_i^{HoL}을 찾아 지연규격을 충족, 즉 $t_R(P_i^{HoL}$전송시작 직전)$\geq F_i^{HoL}$이면 전송하고, 아니면 이를 폐기한 후 다시 이러한 과정을 반복한다.

Fig. 1의 예를 들어 이러한 패킷 폐기동작을 살펴본다. 각각 ΔT_1, ΔT_i 및 ΔT_n의 타임스탬프 조정값을 갖는 흐름 1, i 및 n가 있으며 이들의 HoL 패킷 P_1^{HoL}, P_i^{HoL} 및 P_n^{HoL}의 타임스탬프는 모두 t_p 값을 갖고 있다. GLWFQ 스케줄러가 t_p 시점에 이들 패킷을 출력링크로 전송하려고 한다. 그러면 이들 흐름의 타임스탬프 조정값에 의해 패킷들의 우선 순위값이 결정된다. $\Delta T_1<\Delta T_i<\Delta T_n$일 경우 P_1^{HoL}이 선택되고 $t_R(P_1^{HoL}$전송시작 직전)$=t_p\leq F_1^{HoL}$을 만족하여 이 패킷이 출력링크로 전송된다. P_1^{HoL} 전송 후 P_i^{HoL}, P_n^{HoL} 순서로 선택되나 이들 두 패킷은 모두 지연규격 위반으로 폐기된다.

(5)에서 $\Delta T_i=0$ 이면 우선순위는 바로 F_i^{HoL}가 된다. F_i^{HoL}은 인가되는 부하에 종속되는 확률변수로서 패킷크기가 0에 근접하는 GPS의 경우 저부하 상태이면 남는 자원을 이용해서 미리 전송하므로 $F_i^{HoL}>$

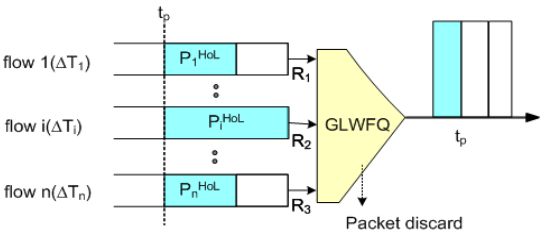

Fig. 1. Packet Discard Concept Diagram.

t_p가 되나 과부하 상태이면 $F_i^{HoL}<t_p$, 100% 부하이면 $F_i^{HoL}=t_p$가 된다. 패킷 크기가 0보다 큰 WFQ의 경우 역시 과부하 상태에서 $F_i^{HoL}<t_p$ 조건이 발생하며 이때 지연규격 위반으로 패킷을 폐기하므로 패킷 손실이 발생한다. WFQ 스케줄러는 작업보존 시스템[4]으로서 출력링크 자원의 낭비 없이 100% 능력으로 부하를 처리한다. 예를 들어 출력링크 용량의 105%의 과부하가 인가될 경우 100%를 처리하므로 남은 5%의 부하가 처리되지 못하고 이들에 해당하는 패킷들에 지연규격 위반이 발생한다. 흐름들의 트래픽 특성은 상호 독립적이므로 지연규격 위반 현상이 흐름의 특성과 상관없이 모든 패킷들에 동일한 확률로 발생하며, 임의의 흐름 i의 HoL 패킷에 대해 $F_i^{HoL}<t_p$가 될 확률은 바로 과부화 비율과 동일하다. (5)에서 0이 아닌 $\triangle T_i$가 더해질 경우 이는 결정변수로 작용하므로 그 만큼 U_i^{HoL}값을 증가시킨다. 따라서 $\triangle T_i$가 클수록 흐름 i의 우선순위는 그만큼 낮아져 P_i^{HoL}의 스케줄링이 늦어진다. 이 결과 지연규격 위반의 가능성이 높아져 패킷 손실률이 증가하게 된다. 이로부터 다음 정리 1이 도출된다.

정리 1 : 흐름의 패킷 손실률은 흐름의 트래픽 특성과 요구 서비스 품질에 상관없이 흐름의 타임스탬프 조정값에 의해서 결정된다.

3.2 GLWFQ 설계

정리 1로부터 흐름의 패킷 손실률은 타임스탬프 조정값에만 의존하므로 다음과 같이 어떤 패킷에 대해 조정이후 타임스탬프값 T_a는 조정이전 타임스탬프값 T_b에 타임스탬프 조정값 $\triangle T$를 더한 값으로 계산한다.

$$T_a = T_b + \triangle T \tag{6}$$

(3)과 같이 타임스탬프는 패킷의 크기를 예약속도로 정규화 한 가상 패킷길이 단위를 갖는다. 타임 스탬프 조정값은 타임스탬프 관리에 있어 혼란을 방지하기 위해 스케줄러에서 가장 작은 가상 패킷길이인 $L_{virtual}^{\min}$을 초과하지 않도록 한다.

(6)에 의해 결정되는 조정된 타임스탬프 값은 흐름의 트래픽 특성과 요구 서비스 품질에 상관없이 오로지 타임스탬프 조정비율에 의해서 결정되므로 일반 트래픽 환경에 적용할 수 있다. 흐름별 차별적인 패킷 손실률을 보장할 수 있는 최적 타임스탬프 조정값 $\triangle T$를 구하는 문제는 3.3장에서 다룬다.

GLWFQ 스케줄러의 동작은 다음 전송할 패킷을 선택할 때 타임스탬프 조정값으로 각 패킷의 타임스탬프 값을 조정하는 것을 제외하고는 [10]과 동일하다. 먼저 흐름별 차별적인 패킷 손실률 요구로부터 전체 트래픽에 대한 패킷 손실률을 계산하고, 시뮬레이션을 통해 주어진 전체 패킷 손실률로부터 허용 누적 가중치 ψ^L를 구하여 과잉 흐름 수락제어에 사용한다. 패킷 도착시 스케줄러는 그 패킷의 가상 전송 종료 시각을 계산하여 패킷의 타임스탬프 영역에 기입하고, 서버 가상시간을 갱신한다. 전송 중인 패킷이 전송 완료될 때 스케줄러는 각 흐름 큐의 HOL (Head-of-Line) 패킷의 타임스탬프 값을 그 흐름의 타임스탬프 조정값으로 강제 조정한 이후 가장 앞선, 즉 가장 작은 값을 갖는 패킷을 찾아 전송하고 서버 가상시간을 갱신한다.

3.3 최적 타임스탬프 조정값 구하기

먼저 타임스탬프 조정값이 패킷 손실률에 끼치는 영향을 분석한다. 스케줄러가 취급하는 전체 트래픽에 대한 트래픽 손실률을 스케줄러 패킷 손실률이라 하고 L_{GLWFQ}로 표시한다. WFQ 스케줄러는 작업보존시스템이고 스케줄러에 수용되는 각 흐름의 타임스탬프 조정 값으로 그 흐름 패킷의 타임스탬프를 조정하더라도 스케줄러에 인가되는 트래픽 부하의 량에는 아무런 변화가 없으므로 다음 정리 2가 성립한다.

정리 2 : L_{GLWFQ}는 가해진 트래픽 부하의 량에 영향을 받을 뿐 각 흐름의 타임스탬프 조정 값과 무관하다.

흐름의 타임스탬프 조정값이 클수록 (6)에 의해 계산되는 우선순위가 낮아진다. 그 결과 그 흐름의 패킷이 스케줄러에서 대기하는 시간이 길어져 지연규격 초과로 인해 폐기되는 패킷의 수가 늘어난다. 반대로 흐름의 타임스탬프 조정값이 작을수록 패킷 전송에 있어 우선순위가 높아지고, 그 결과 그 흐름의 패킷이 스케줄러에서 대기하는 시간이 짧아져 지연규격 초과로 인해 폐기되는 패킷의 수가 줄어든다. 이로부터 다음 정리 3이 성립한다.

정리 3 : 임의의 흐름에 대하여 $\triangle T$가 클수록 패킷 손실률이 높아지고, $\triangle T$가 작을수록 패킷 손실률이 낮아진다.

임의의 흐름 i의 타임스탬프 조정값 $\triangle T_i$를 높이면 정리 3에 의해 흐름 i의 패킷 손실률이 높아진다. 하지만 정리 2에 의해 스케줄러의 패킷 손실률은 일정하므로 흐름 i의 패킷 손실률이 높아진 만큼 다른 흐름의 패킷 손실률은 낮아지게 된다. $\triangle T_i$를 낮추면 반대의 현상이 일어난다. 이로부터 다음 정리 4가 도출된다.

정리 4 : 임의의 흐름의 $\triangle T$를 키우면 그 흐름의 패킷 손실률이 높이지는 반면 다른 흐름의 패킷 손실률은 낮아지고, 반대로 임의의 흐름의 $\triangle T$를 낮추면 그 흐름의 패킷 손실률이 낮아지는 반면 다른 흐름의 패킷 손실률은 높아진다.

참고로 정리 2, 3 및 4는 [10]의 결과를 일반 트래픽 환경에 맞도록 수정한 것이다.

허용 손실률이 동일한 흐름이 다수 존재할 수 있으므로 이들 집단을 하나의 클래스로 묶고 허용 패킷 손실률이 증가하는 순서로 클래스 번호를 할당한다. 클래스 c의 트래픽 량과 목표 패킷 손실률 및 전체 클래스의 수를 각각 V_c , O_c 및 N으로 표기한다. 그러면 클래스 c의 패킷 손실량 L_c는 간단히 V_cO_c로 계산된다. 클래스들의 타임스탬프 조정값과 그들의 패킷 손실률의 상호관계를 살펴본다. 임의의 클래스 c의 타임스탬프 조정값 $\triangle T_c$를 높이면 정리 3에 의해 클래스 c의 패킷 손실량 L_c가 늘어난다. 하지만 정리 2에 의해 스케줄러의 패킷 손실률은 일정, 즉 전체 패킷 손실량은 일정하므로 늘어난 패킷 손실량 만큼 다른 클래스의 패킷 손실률은 낮아지게 된다. $\triangle T_i$를 낮추면 반대의 현상이 일어난다. 이로부터 다음 정리 5가 도출된다.

정리 5 : 임의의 클래스 c의 $\triangle T_c$를 높이면 클래스 c의 L_c가 늘어나고, 늘어난 L_c 만큼 다른 클래스의 패킷 손실률은 낮아지게 되며, 낮아지는 패킷 손실률은 늘어나는 L_c 량에 비례한다. $\triangle T_c$를 낮추면 반대의 현상이 일어난다.

클래스 c 내 흐름의 수를 N_c, 클래스 c내 흐름 i의 패킷 크기와 초당 발생 패킷 수를 각각 L_c^i와 S_c^i로 표기하자. 그러면 L_{GLWFQ}는 다음과 같이 구할 수 있다.

$$L_{GLWFQ} = \frac{\sum_{c=1}^{N} V_c O_c}{\sum_{c=1}^{N} V_c} \text{ 및 } V_c = \sum_{i=1}^{N_c} L_c^i S_c^i \tag{7}$$

한편 실제 트래픽 환경에서 볼 때 허용 패킷 손실률이 낮은 흐름은 그 만큼 서비스 요금이 비싸지므로 허용 패킷 손실률이 높지만 요금이 저렴한 대중적 서비스가 더 많이 이용될 것이다. 따라서 허용 패킷 손실률이 높을수록 트래픽의 량이 더 많아지는 경향이 있다. [10]은 패킷 손실률이 증가하는 클래스 순서로 타임스탬프 조정값을 찾는 일종의 오름차순 알고리즘을 사용하였다. 즉 트래픽 량이 적고 허용 패킷 손실률이 낮은 클래스의 타임스탬프 조정값을 먼저 찾은 후 트래픽 량이 많고 허용 패킷실률이 높은 클래스의 타임스탬프 조정값을 구한다. 허용 패킷 손실률이 높은 클래스일수록 더 큰 타임스탬프 조정값이 필요한데다 트래픽 량까지 많은 경향이 있으므로 이들의 타임스탬프 값 조정으로 인해 허용 패킷 손실률이 낮고 트래픽 량이 상대적으로 적은 클래스가 그만큼 더 큰 영향을 받는다. 이 결과 이들 클래스의 최적 타임스탬프 조정값을 찾는데 긴 시간이 소요될 수 있다. 이러한 문제를 해결하기 위해 본 논문에서는 패킷 손실률이 감소하는 순서로 클래스의 타임스탬프 조정값을 찾는 일종의 내림차순 알고리즘을 제시한다.

클래스 c의 패킷 손실률 측정치를 M_c로 표기한다. 그리고 클래스 k부터 j까지 $(k-j+1)$개의 클래스를 하나로 묶은 슈퍼 클래스를 클래스 $(k \sim j)$로 표기한다. 그러면 $O_{N^{\sim}1}$와 $M_{N^{\sim}1}$는 바로 전체 트래픽 묶음에 대한 목표 패킷 손실률과 패킷 손실률 측정치가 된다.

클래스 N의 타임스탬프 조정값 $\triangle T_N$부터 내림차순으로 $\triangle T_{N-1}$ … 및 $\triangle T_1$ 값을 찾는 Fig. 2에 도시된 알고리즘을 기술한다. 먼저 클래스 $(c-1 \sim 1)$을 클래스 $(c-1)$과 클래스 $(c-2 \sim 1)$로 분리한 후 $\triangle T_{c-1^{\sim}1}$값을 조정하여 $M_c = O_c$가 되게 한다. 그리고 $M_{c-1} = O_{c-1}$가 되도록 $\triangle T_{c-2 \sim 1}$값을 조정하면 $\triangle T_{c-1 \sim 1}$값이 바로 $M_{c-1} = O_{c-1}$이 되는 그러한 $\triangle T_{c-1}$값에 근접한다. 여기서 $\triangle T_N$의 초기값은 타임스탬프 조정 최대값의

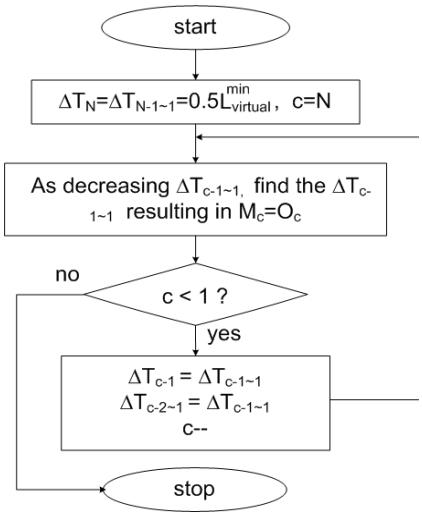

Fig. 2. Timestamp adjustment initial value algorithm.

중간인 $0.5*L_{virtual}^{\min}$로 정하고, M_c=O_c가 되는 그러한 $\Delta T_{c-1\sim1}$값은 시뮬레이션을 통해 찾는다. 이렇게 구한 값은 최적 타임스탬프 조정값을 찾기 위한 초기값으로 사용한다.

이제 타임스탬프 조정 초기값으로부터 최적 조정값을 구하는 과정인 Fig. 3의 알고리즘을 살펴본다. 클래스 N부터 1까지 내림차순으로 각 클래스의 타임스탬프 조정값을 감소시키면서 그의 목표 패킷 손실률과 패킷 손실률 측정치가 일치하는 그러한 타임스탬프 조정값을 찾는다. 정리 5에 의해 낮은 클래스의

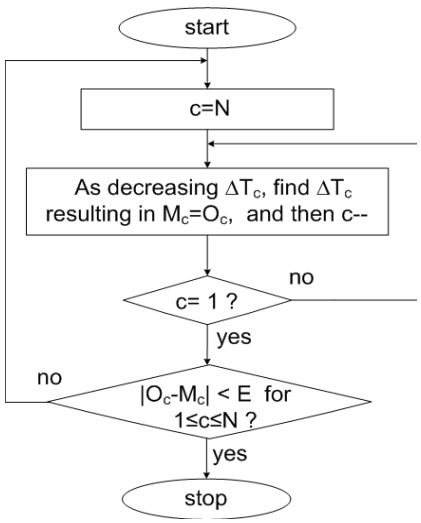

Fig. 3. Optimal timestamp adjustment value algorithm.

타임스탬프 조정값을 낮추면 높은 클래스의 패킷 손실률 측정치가 높아지지만 이러한 과정을 반복할수록 높아지는 정도가 줄어들어 최적값에 수렴하게 된다. 따라서 모든 클래스에 대해 목표치에 대한 측정값의 허용 오차인 E값 범위 내에 들어갈 때까지 이러한 과정을 반복 수행하여 각 클래스의 최적 타임스탬프 조정값을 구할 수 있다.

4. 성능평가

흐름별 별도의 큐가 필요한 WFQ는 규모성(scalability) 문제로 인해 아직 대규모 스위치에는 적용되기 힘들므로 100여 VoIP호를 동시에 수용할 수 있는 소규모 스위치를 대상으로 시뮬레이션을 통해 성능을 평가하였다. 시뮬레이션은 SMPL(Simulation Model Programming Language)에 WFQ와 관련 프로그램을 추가하여 실시하였다. VoIP 트래픽은 대부분 음성이므로 흐름별 요구 속도가 거의 같고, RTP (Realtime Transport Protocol)에 의해 패킷이 수송되므로 패킷 크기 또한 유사하다. 하지만 흐름별 종단간 허용 지연이 다르고 흐름별 패킷 전달 경로 또한 서로 달라 각 노드에 할당되는 흐름별 지연규격은 큰 차이를 보일 수 있다. 따라서 본 논문에서는 노드에서의 지연규격이 각각 4, 5 및 6ms인 유형 1, 2 및 3의 세가지 유형의 흐름을 대상으로 실험하였다. 이들 유형은 모두 목표 패킷 손실률이 각각 0.1, 0.5, 1, 2 및 3%인 5가지 클래스로 분류되며 클래스별 흐름의 수가 균등하게 분포하는 균등분포와 지수적으로 분포하는 지수분포의 두 가지 부하를 사용하였다.

먼저 클래스별 수용 흐름 수가 유형 별로 7개씩 총 21개로 동일한 균등분포 부하의 105 VoIP 호에 대하여 기존 방식인 [9] 및 [10]과의 비교 측면에서 성능을 평가하였다. 호의 통화시간은 모두 1시간으로 하였다. 모든 클래스에 대해 평균 패킷 손실률의 목표치에 대한 측정치의 오차 범위 E를 5%로 하였다. 참고로 흐름 수가 클래스별로 균등분포하므로 (7)에 의해 스케줄러 패킷 손실률은 1.32%로 계산된다.

본 논문에서 제안된 내림차순 알고리즘하에서 라운드, 즉 알고리즘의 수행 횟수에 따른 타임스탬프 조정값의 수렴과정을 Fig. 4에 도시하였다. 3 라운드만에 5% 오차 범위내의 패킷 손실률 측정치를 보장해주는 타임스탬프 조정값을 얻을 수 있었다. 첫 번

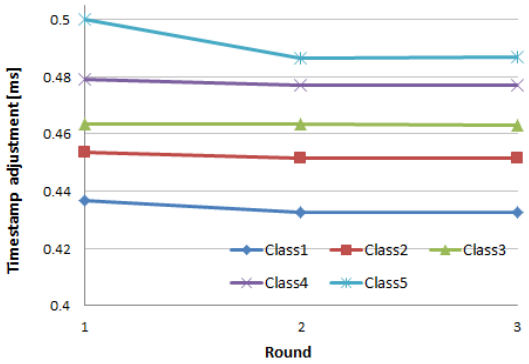

Fig. 4. Timestamp adjustment value under the decreased-algorithm.

째 라운드는 Fig. 2의 초기값 도출 알고리즘을 수행한 것이고 두 번째 라운드부터는 Fig. 3의 알고리즘을 반복 수행한 것이다. Fig. 5는 [10]의 오름차순 알고리즘하에서의 수렴과정을 보여준다. 기존방식과의 비교시 수렴시간이 5회에서 3회로 대폭 단축됨을 보여준다.

최적 타임스탬프 조정값 하에서 측정된 유형별 평균 패킷 손실률을 Table 1에 도시하였다. 유형별로 괄호속에 표기된 각 클래스의 목표 패킷 손실률에 근접한 실측값을 보여 요구된 패킷 손실률을 보장할 수 있음을 관찰할 수 있다. 클래스별 평균은 5% 이내의 오차범위에 들어가나 유형 3의 클래스 2와 유형 2의 클래스 4와 같이 일부 유형의 경우 이를 벗어나는데, 이는 제한된 시간(1시간)과 적은 흐름 수(7개 흐름)의 제약으로 인해 확률적으로 일어날 수 있는 부분으로 설명할 수 있다.

GLWFQ 스케줄러의 경우 105개의 흐름을 수용할 수 있는 반면 기존방식 [9]는 97개의 흐름만 수용할

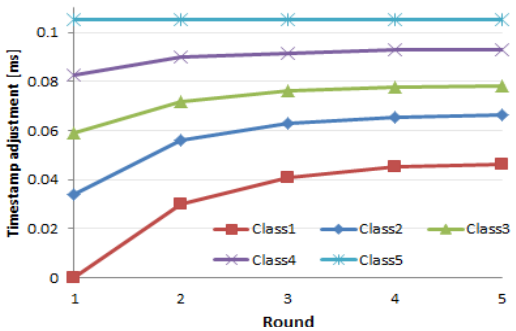

Fig. 5. Timestamp adjustment value under the increased-algorithm.

Table 1. Measured packet loss under uniform scenario

Class	1(0.1)	2(0.5)	3(1.0)	4(2.0)	5(3.0)
Average	0.098	0.503	1.02	2.02	3.02
Type 1	0.102	0.493	1.01	1.99	3.02
Type 2	0.097	0.481	0.98	2.11	2.99
Type 3	0.095	0.525	1.07	1.96	3.03

Table 2. Measured packet loss under exponential scenario

Class	1(0.1)	2(0.5)	3(1.0)	4(2.0)	5(3.0)
Average	0.099	0.497	1.01	2.02	3.01
Type 1	0.106	0.490	0.98	2.06	2.95
Type 2	0.098	0.487	0.96	1.98	3.02
Type 3	0.094	0.515	1.08	1.99	3.06

수 있었다. 이는 가장 엄격한 패킷 손실률이 0.1%이므로 손실률을 차별화 할 수 없는 기존방식의 경우 3%의 느슨한 패킷 손실률의 클래스 5 까지도 0.1%의 엄격한 손실률을 제공해야 하기 때문으로 분석된다. 따라서 기존방식과의 비교시 8%의 성능개선을 얻을 수 있었다.

다음은 지수분포 부하에 대해 살펴본다. 클래스 1부터 수용 흐름 수를 3, 6, 12, 24 및 48로 할당하고, 유형별로 균등 분포하는 전체 93개 VoIP 호의 한 시간 통화분에 대해 성능평가를 하였다. 스케줄러 패킷 손실률은 (7)에 의해 2.23%로 계산된다. 구해진 최적 타임스탬프 조정값 하에서 측정된 유형별 평균 패킷 손실률을 Table 2에 도시하였다. 균등분포 부하에 대한 Table 1과 마찬가지로 유형별로 클래스의 목표 패킷 손실율에 근접한 실측값을 보여준다. 이로부터 요구된 패킷 손실률을 보장할 수 있음을 관찰할 수 있다.

GLWFQ 스케줄러의 경우 93개의 흐름을 수용할 수 있는 반면 기존방식 [9]는 84개의 흐름만 수용할 수 있었다. 이는 모든 클래스에 대해 0.1%의 엄격한 손실률을 보장할 해야 하기 때문이다. 따라서 기존방식과의 비교시 11%의 성능개선을 얻을 수 있었다.

5. 결 론

손실 WFQ에서 흐름의 패킷 손실률이 트래픽 흐름의 특성과 요구 서비스 품질에 상관없이 타임스탬

프 조정값에 의해서 결정됨을 밝혀냈고, 이러한 성질을 이용하여 일반적인 환경에서 적용할 수 있는 차별화된 패킷 손실률을 보장할 수 있는 손실 WFQ 스케줄러를 제안하였다. 그리고 요구된 패킷 손실률을 보장하기 위해 필요한 최적 타임스탬프 조정값을 구하는 방법에 있어 종래의 오름차순 알고리즘 대신 수렴속도가 빠른 내림차순 알고리즘을 제안하였다.

성능 고찰 결과 3회의 최적 타임스탬프 조정값 알고리즘 수행으로 패킷 손실률 목표치 대비 5% 오차범위내의 측정값을 얻을 수 있는 타임스탬프 조정값을 얻을 수 있었다. 5회의 알고리즘 수행이 필요했던 종래의 오름차순 알고리즘에 비해 40%의 수렴시간 단축 효과를 얻을 수 있었다. 지연 규격이 상이한 3가지 유형의 트래픽에 대해 실험한 결과 트래픽 유형과 상관없이 요구 허용 패킷 손실률을 보장할 수 있음을 확인하였다. 패킷 손실률을 차별화 할 수 없는 이전의 연구결과와 비교시 균등분포 부하의 경우 8%, 지수분포 부하의 경우 11%의 자원 이용도 개선 효과를 관찰할 수 있었다.

REFERENCE

[1] X. Xiao and L.M. Ni, "Internet QoS: A Big Picture," *IEEE Network*, Vol. 13, No. 2, pp. 8-18, 1999.

[2] P.J. Argibay-Losada, A. Suárez-González, C. López-García, and M. Fernández-Veiga, "A New Design for End-to-end Proportional Loss Differentiation in IP Networks," *Computer Networks,* Vol. 54, Issue 2, pp. 1389-1403, 2010.

[3] M. Elhaddad, R. Melhem, and T. Znati, "Supporting Loss Guarantees in Buffer-Limited Networks," *Proceeding of IEEE/ACM International Symposium on Quality of Service*, pp. 239-250, 2006.

[4] A.K. Parekh and R.G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: the Single Node Case," *IEEE/ACM Transactions on Networking*, Vol. 3, No. 1, pp. 344-357, 1993.

[5] A. Demers, S. Keshav, and S. Shenker, "Design and Analysis of a Fair Queuing Algorithm," *Proceeding of ACM Special Interest Group on Data Communication*, pp. 1-12, 1989.

[6] T.J. Kim, "A Preventive Intra-path Load Balancing based on the Probabilistic Characteristics of the Quality of Service," *Journal of Korea Multimedia Society*, Vol. 13, No. 2, pp. 279-286, 2010.

[7] S.Y. Bae, *VoIP Planning and Evaluation through the E-model based Speech Transmission Quality Analysis*, Doctor's Thesis of Kyonggi University, 2004.

[8] D. Zhang and D. Ionescu, "Providing Guaranteed Packet Loss Probability Service in IP/MPLS-based Networks," *Proceeding of IEEE International Conference on Communications*, pp. 5772-5776, 2008.

[9] T.J. Kim, "A Weighted Fair Packet Scheduling Method Allowing Packet Loss," *Journal of Korea Information and Communication Society*, Vol. 35, No. 9, pp. 1272-1280, 2010.

[10] T.J. Kim, "A Packet Loss Differentiation and Guarantee in the Weighted Fair Queuing," *Journal of Korean Institute of Information Technology*, Vol. 11, No. 10, pp. 67-75, 2013.

김 태 준

1980년 2월 경북대학교 졸업
1982년 2월 한국과학기술원 졸업 (석사)
1999년 8월 한국과학기술원 졸업 (박사)
1982년~1996년 ETRI 근무

2014년 12월 현재 공주대학교 교수
관심분야: 인터넷 엔지니어링