A RSVP-capable Router to improve the bandwidth utilization efficiency in resource reservation

Tae Joon Kim

ABSTRACT

SVP−capable router supporting guaranteed services on the internet generally uses a packet scheduler based on the Weighted Fair Queuing (WFQ) algorithm to secure required qualities of traffic flows. In this paper we proposed a RSVP−capable router based on the LOFQ (Latency optimized fair queuing) algorithm that improves the efficiency in bandwidth utilization with keeping the compatibility with the original RSVP procedure. The proposed router reserves the optimal amount of resource for each flow to secure its required quality−of−service with the recently introduced LOFQ scheduler. The results of the simulation applying the proposed router to an evaluation network showed that it may yield the gain of up to 30% compared to that in the original one in terms of the number of admitted flows.

Key words: WFQ(WFQ), QoS router(QoS 라우터), resource reservation(자원예약), RSVP(RSVP), LOFQ(LOFQ), RSVP−capable router(RSVP 라우터), Bandwidth utilization(대역폭 이용 효율성)

1. 서 론

인터넷 전화, 인터넷 영상회의와 같은 고 수준의 서비스 품질을 요구하는 실시간 멀티미디어 통신 서비스를 사용하기 위해 IETF(Internet Engineer Task Force)에서 종합서비스(IntServ) 모델을 제시하였고[1], 자원예약을 위해 RSVP(Reservation protocol)을 규격화 하였다[2]. IntServ 모델에서 품질 보장형 서비스를 지원하기 위해 자원예약 기반의 RSVP 라우터를 사용하며 RSVP 라우터는 트래픽 흐름의 요구 속도를 보장하고 허용되는 레이턴시(latenecy), 즉 지연규격을 준수하기 위해 QoS
(Quality-of-Service) 스케줄러를 탐색한다. 이러한 품질 보장형 서비스에 요구되는 기술분야는 RSVP 초기와 QoS 스케줄러의 두 영역으로 나눌 수 있다.


한편 유체모델 기반의 이상적 패킷 스케줄링 방식인 GPS(General Processor Sharing)[10]가 WFQ(Weighted Fair Queuing)[11]에 의해 구현되었다. WFQ는 호스트간의 상호 간섭을 차단하여 각 호스트의 품질 특성이 다른 호스트로부터 영향을 받지 않도록 하는 호스트 분리와 차별화된 품질 보장을 지원하는 동 공정 큐용의 요구사항을 충실히 따르기 때문에 IETF IntServ 모델의 RSVP-라우터에 적용되었으나[12]. 그러나 호스트의 스케줄링 속도를 높일 경우 스케줄링 속도와 요구속도의 차이만큼 속도(대역폭)차이가 발생하므로, 특히 인터넷 전화와 같이 트래픽의 속도는 낮지만 허용한 중간단 지연특성을 요구하는 트래픽 호스트의 경우 대역폭 이용도의 저하가 심각해짐을 막아야한다[13]. 그 동안 WFQ의 대역폭 저하 문제는 WFQ의 속도와 대역폭의 결합특성 때문이라고 이해되어 왔다[14-15]. 최근에 WFQ의 고정적인 문제의 근본은 레이터링이 최적화되지 않아서 발생하는 지연차원의 낭비 때문로 밝혀졌고, 그러한 낭비를 방지하기 위해 레이터링이 최적화되는 LOFQ(Latency-Optimized Fair Queuing) 방식이 연구되었다[16,17].

본 연구는 RSVP 기반의 자원예약에 있어 기존 RSVP 철차와 호환성을 유지하면서 네트워크 대역폭 자원의 이용 효율을 높일 수 있는 LOFQ 기반의 RSVP-라우터에 관한 것이다. 제안하는 라우터는 LOFQ 스케줄러를 탐색하며, 기존 RSVP-라우터의 RSVP 모듈과 수락 제어 모듈의 기능을 일부 수정한 다. 그리고 제안한 RSVP-라우터를 평가 네트워크에 적용하여 시뮬레이션을 통해 성능특성을 비교 평가한다.

본 논문의 구성은 다음과 같다; 2장에서 LOFQ 스케줄러를 상세히 소개하고, 3장에서 제안한 LOFQ 기반의 RSVP-라우터를 상세히 기술한다. 4장에서 LOFQ 스케줄러의 성능 특성을 분석한 후 5장에서 평가 네트워크를 대상으로 LOFQ 기반 RSVP-라우터의 성능을 기존의 방식과 대비하여 비교 평가한다. 마지막으로 6장에서 결론을 끝낸다.

2. LOFQ 스케줄러 소개

스케줄러에 도착하는 호스트의 속성은 호스트 규격 F(r,b,M)를 정의할 수 있다. 여기서 r은 호스트의 요구 속도, b는 호스트의 중간단 지연한계 중 스케줄러에 배분되는 지연한계(delay budget), M은 호스트의 최대 패킷크기가다. 여기서 지연한계가 바로 스케줄러가 준수해야 하는 호스트의 지연규격이다. WFQ에서 호르 구식 F(r,b,M)는 얻는 얻의 호름의 고유 레이터시 q는 다음과 같이 주어진다[11].

\[ q = \frac{M}{r} \frac{c_{max}}{C_{b}}. \]  \hspace{1cm} (2.1)

(2.1)에서 C는 레이터링의 용량으로 스케줄러의 총 대역폭을 의미하고, M_{max}는 모든 호름의 최대 패킷크기이다. WFQ는 요구속도와 지연규칙을 동시에 만족시켜야 하므로 얻의 호름의 스케줄링 속도 s는 (2.1)로부터 다음과 같이 계산되었다. 참고로 스케줄링 속도를 얻의 속도 또는 예약대역폭이라 한다.

\[ s = \max (r, b), \text{여기서} \quad r' = \frac{MC_b}{C_{b} - M_{max}}. \]  \hspace{1cm} (2.2)

(2.2)에서 r'는 제한 요구속도라 한다. 그러면 WFQ에서 얻의 호름의 실제 레이터시 q는 다음과 같이 표현된다.

\[ q = \frac{M}{s + \frac{C_{max}}{C_{b}}} \leq b. \]  \hspace{1cm} (2.3)

임의 호름의 고유 레이터시 q와 지연규칙 b는
(2.1)과 (2.3)로부터 \( q' = b, q'' > b \) 또는 \( q' < b \)의 관계를 갖는다. \( q' = b \)의 경우 저장의 손실 없이 호름을 수용할 수 있지만 \( q'' > b \)의 경우 스케줄링 속도를 높여서, 즉 파업에 의점을 통해서 저장공격을 줄여, 즉 \( q'' > b \)가 되게 해야 한다. 이때 스케줄링 속도와 요구속도의 차이를 만큼 대역폭 자원이 소실되지만 저장공격을 만족시키기 위해 피할 수 없다. 한편 \( q' < b \)의 경우 \( q'' = b \)가 되게 하려면 스케줄링 속도를 요구속도보다 낮게 해야 한다. 하지만 호름의 요구속도를 보장해야하기 때문에 스케줄링 속도를 낮추지 않으면 저장공격보다 더 엄격한 테이터스를 갖게 된다. 이 결과 \((b-q)\)만큼의 저장 자원이 낭비되어 스케줄러 사용 이용 효율이 저하된다. 이러한 낭비되는 저장자원을 재활용하여 대역폭 이용도를 개선한 LOFQ가 연구되었다(16,17).

[16,17]의 내용을 소개하면 다음과 같다. LOFQ는 레이터시 저수 \( b \geq 0 \)과 \((D,b)\)의 \( D \)는 호름 중 가장 엄격한 저장공격 값으로 설정됨, 어떤 값을 갖는 레이터시 바이어스 \( b \)를 도입한 후 (2.4)과 같이 각 호름의 레이터시 \( q \)를 \( b \)와 스케줄링 속도 \( h \)의 함수가 되게 하였다. 그리고 \( b \)와 \( h \)를 조정하여 \( q \)를 \( b \)와 동일하게, 즉 항상 저장공격과 동일한 최적의 레이터시를 갖도록 하여, 대역폭 종류를 최소화하기 위해 \( h \)를 우선적으로 증가한다. 참고로 \( h \)만 바로 그 호름이 점유하는 대역폭이 된다.

\[
q = D_{b} + \frac{M}{h} = b. \tag{2.4}
\]

레이터시 바이어스 \( D \)는 호름을 많이 수용할수록 증가하지만 그 값이 \( D_{b} \)를 초과할 수 없다. 따라서 대역폭 \( (BW: Bandwidth) \)와 마찬가지로 레이터시 바이어스는 일종의 자원으로 취급될 수 있으므로 이를 저장자원 \( (DW: Delay Width) \)이라고 한다. 그리고 \( D_{b} \)와 \( D \)를 각각 스케줄러의 \( DW \)용량과 스케줄러의 점유 \( DW \)와 한다. 그리고 \( b \)의 레이터시 저수를 갖는 호름에 의해 증가되는 레이터시 바이어스 값, 즉 그 호름의 점유 \( DW \)가 되는 \( d \)는 다음과 같이 주어진다.

\[
d = \max(0,1-\beta M/C_{b}). \tag{2.5}
\]

그리고 스케줄러의 점유 \( DW \), 즉 레이터시 바이어스 \( D \)는 다음과 같이 주어진다.

\[
D = \max(\frac{M_{\text{max}}}{C_{b}}, \sum_{b}(\max(0,1-\beta M/C_{b})). \tag{2.6}
\]

여기서 \( b(t) \)는 수용된 호름 집합이다.

(2.4)과 (2.5)로부터 호름의 점유 \( BW \)와 점유 \( DW \)도는 서로 반비례하므로 점유 \( BW \)를 늘리는 대신 점유 \( DW \)를 줄일 수 있고, 반대로 점유 \( DW \)를 늘리는 대신 점유 \( BW \)를 줄일 수 있는, 즉 \( BW \)와 \( DW \)가 상호 변환될 수 있음을 알 수 있다. 여기서 저수를 B2D 변환, 후자를 D2B 변환이라고 한다. 그리고 늘려 나가는 \( BW \) 대비 줄어지는 \( DW \)의 비율을 B2D 변환 효율 \( E_{b2d} \), 줄어나는 \( DW \) 대비 늘어지는 \( BW \)의 비율을 D2B 변환 효율 \( E_{d2b} \)라고 한다. WFQ의 경우 스케줄러의 점유 \( BW \)가 \( C_{b} \)에 도달하면 더 이상 호름을 수용할 수 없다. 하지만 LOFQ의 경우 다음과 같이 \( BW \)의 변환을 통해 대역폭 이용도를 높이다; \( DW \)는 부족하면 \( BW \)는 충분할 경우 B2D 변환을 통해 부족한 \( DW \)를 확보하고, 반대로 \( BW \)는 부족하면 \( DW \)는 충분할 경우 D2B 변환을 통해 부족한 \( BW \)를 확보한 후 그 호름을 수용한다.

LOFQ 스케줄러는 호름수락제어, 점유자원 최적화 및 폐킷 스케줄러의 세 가지 기능으로 구성된다. 호름 수락제어 기능은 필요한 자원이 있으면 호름을 수락하며 자원이 부족할 경우 자원변환 또는 점유자원 최적화를 통해 필요한 자원을 확보한다. 호름마 다 고유 \( E_{d2b} \)와 \( E_{b2d} \)를 가짐으로 호름별로 보다 효율이 좋은 자원을 사용하도록 함으로서 자원 사용량을 줄일 수 있다. 점유자원 최적화는 이러한 특성을 이용하여 점유자원을 최소화한다. 폐킷 스케줄러 기능은 폐킷이 도착할 때 태임스테프를 계산한 후 호름 큐에 수용되는 폐킷 큐생 기능과 호름의 큐에 대기중인 폐킷을 그들의 태임스테프 값이 작은 순으로 출력링크를 통해 전송하는 폐킷 큐 서비스 기능으로 구성된다.

임의 호름 \( i \)의 k번째 폐킷 \( P_{k} \)의 태임스테프 \( T_{k} \)는 다음과 같이 계산한다.

\[
T_{k} = \max(S_{k}^{i}, (v(t)) + \min(1, b_{i}) m_{k} h_{i}). \tag{2.7}
\]

여기서 \( S_{k}^{i} \)는 \( k \)번째 폐킷의 전송 시작시간, \( m_{k} \)는 \( P_{k} \)의 크기, \( h_{i} \)는 호름 \( i \)의 스케줄링 속도, \( v(t) \)는 수신 가능한 시간이다. 호름의 태임스테프가 그의 레이터시 저수 \( b_{i} \) 값에 따라 폐킷의 전송 시작시점 \( \text{Start-time} \)부터 종료시점 \( \text{Finish-time} \)까지 임의의 값을 가질 수 있어 LOFQ를 GFQ(General-time Fair Queuing)라고도 부른다(16). 참고로 \( b_{i} \)가 1일 경우
LOFQ는 바로 종료시점 기반 타임스탬프 계산방식을 적용하는 WFQ가 된다.

3. LOFQ 기반의 RSVP-라우터

3.1 기존 RSVP 자원예약과 문제점

먼저 제안방식의 근간이 되는 기존의 RSVP 절차와 자원 할당 과정을 그림 1을 통해 살펴본다. 단말 $a_1$에서 $a_2$로 향하는 단방향 흐름의 자원예약을 위해 단말 $a_1$이 원하는 흐름 규격 정보, 즉 Tspec를 담은 PATH 메시지를 발송한다. 이 메시지는 RSVP-라우터 0, 1, 2 및 3을 거쳐 단말 $a_2$에 도착한다. PATH 메시지를 전달 받은 경로상의 각 RSVP-라우터는 PATH 메시지의 Adspect에 $C_{aw}$와 $D_{aw}$ 영역에 자신의 C값과 D값을 각각 더한 후 다음 단으로 넘긴다. C와 D는 해당 홀(hop)에서 발생하는 흐름의 지연을 결정하는 상수로 최대 홀 지연은 $(C/R+D)$로 계산된다.[1]. 여기서 R은 흐름에 할당되는 종단간(end-to-end) 예약속도이고 $D$는 라우터에서의 지연인 $D_1$과 다음 단 라우터까지의 경로상에서 지연되는 $D_2$의 합이다. 참고로 라우터에서의 최대지연, 즉 레틴시(q)는 다음과 같다.

$$q = C/R + D_1$$  (3.1)

PATH 메시지를 전달받은 최단단 $a_2$는 종단간 지연한계를 충족하기 위해 필요한 종단간 트래픽 흐름의 예약속도 $R$을 계산한다. R을 계산하기 위해 RSVP는 토큰-버킷(token-bucket) 모델을 사용한다.[2]. 이 모델에 의하면 최대 속도 $p$의 트래픽 흐름은 버킷에 담긴 후 토큰 속도로 트래픽이 세이징(shaping)되어 네트워크로 인가된다. 일반적으로 발신단말이 비트열을 흐름으로 조립한 후 전송하므로 버킷의 크기는 바로 그 흐름의 최대 흐름 크기 $M$이 되고, 트래픽이 토큰 속도로 네트워크에 인가되므로 트론속도가 바로 그 흐름의 요구속도 $r$이 된다. 따라서 종단간 최대 지연 $q_{aw}$은 다음과 같이 계산된다.[1].

$$q_{aw} = \frac{M + C_{aw} + D_{aw}}{R} \cdot \frac{R}{D_{aw} - M(p - R)} \cdot \frac{M + C_{aw} + D_{aw}}{R} \cdot \frac{R}{D_{aw} - M(p - R)}$$  (3.2)

(3.2)에서 $D_{aw}$는 버킷 길이를 의미한다. 흐름의 요구하는 종단간 지연한계를 만족시키는 최소 종단간 예약속도를 종단간 임계 예약속도로 하고 $R^c$로 표기하자. $R \geq p \geq r$인 일반적인 경우를 생각하자. 종단간 지연한계 $w$와 요구속도 $r$을 모두 만족시키기 하므로 (3.2)로부터 R은 다음과 같이 계산된다.

$$R = \max(r, R^c), \text{여기서 } R^c = \frac{M + C_{aw}}{w} - D_{aw}.$$  (3.3)

마찬가지의 방법으로 $p > R \geq r$ 또는 $p > r$인 경우도 구할 수 있다. 

R을 계산한 후 R 값을 포함하는 FLOWSPEC을 받은 RESV 페킷을 화면한다. RESV 페킷은 각 RSVP-라우터는 자신에게 R의 속도자원이 있으면 예약하고 없으면 자원예약 실패를 통보한다. 이와 같은 과정을 통해 자원예약이 성공하면 각 RSVP-라우터는 내장된 WFQ 기반 QoS 스케줄러로 그 흐름에 대해 R의 스케줄링 속도로 스케줄링 함으로써 흐름의 요구속도와 라우터에 할당된 그 흐름의 지연 규격을 충족시키게 된다.

위에서 살펴본 RSVP 자원예약 과정에서 종단간 예약속도 $R$을 구하는 식 (3.3)을 살펴보자. 흐름의 요구속도 $r$이 $R$보다 높은 경우 $r > R$이고 $r$보다 낮은 경우 $r < R$일 경우가 발생할 수 있다. $r > R$인 경우 흐름의 종단간 최대 지연이 요구 지연한계보다 더 긴 경우, 즉 $q_{aw} < w$가 되므로 자원자원이 풍부하고, $r < R$인 경우 흐름의 예약속도가 요구속도보다 더 높아 대역폭 자원이 남비될 수 있다. 이러한 자원낭비 현상은 2장에서 살펴본 WFQ 스케줄러의 경우와 마찬가지이다. RSVP 자원 예약과정에서 일어나는 이러한 자원낭비 문제를 해결하는 것이 본 연구의 핵심 내용이다.

3.2 제안방식

본 논문에서 제안하는 LOFQ 기반 RSVP-라우터, 이하 ERSPV(Enhanced RSVP) 라우터,의 전체 구성도는 그림 2와 같다. 기존 RSVP-라우터와 마찬가지로 구성되나 WFQ기반 페킷 스케줄러를 LOFQ 스케줄러로 대체함에 따라 RSVP 모듈과 수학계어 모
들에서 일부 스케줄러 총속 부분이 수정된다. 특히 RSVP 모듈에 스케줄러 적용부 모듈을 추가하여 RSVP 모듈을 패킷 스케줄러와 분리함으로써 RSVP 모듈의 독립성을 확보한다.

ERSVP-라우터의 RSVP 절차와 자원할당 과정을 살펴보자. RSVP 절차 중 PATH 메시지를 생성하여 호출의 중간간 경로를 통해 전달하고 경로상의 각 라우터가 PATH 메시지를 처리하는 과정은 WFQ 스케줄러를 기반으로 하는 기존의 RSVP 경우와 동일하다. 다만 그림 3과 같이 Adspect는 C와 D 값에 자신의 값과 D 값을 더할 때 LOFQ 스케줄러에 대한 C와 D 값이 아니라 WFQ 기반 스케줄러에 대한 값을 사용한다. 각 라우터에서 호출에 할당하는 스케줄링 속도는 바로 중간간 예약속도 R이므로 (2.3)과 (3.1)로부터 C는 호출의 최대 패킷크기 M인 되고 D는 \( \frac{M^{\text{max}}}{C} \)가 된다.

PATH 메시지를 전달받은 채석측은 (3.2)에 따라 중간간 임계 예약속도 \( R^C \)와 중간간 예약속도 R를 모두 계산한 후 이를 RESV 메시지의 FLOWSPEC 객체에 담아 호출의 역 경로를 통해 경로상의 각 라우터에게 알려준다. 이를 위해 그림 4와 같이 R과 속도 S 영역만 갖는 기존 Rspec을 R, S 및 \( R^C \)의 세 영역을 갖도록 확장한다. RESV 메시지의 FLOWSPEC 객체에 첨부된 Rspec을 담은 방법은 그림 4와 같다. \( R^C \)을 추가하기 위해 Length 영역을 기존 10에서 11로 증가시키고 12번째 워드(Word)에 \( R^C \)값을 수록한다. 이렇게 함으로서 기존 RSVP-라우터는 R 값을 사용하고 제안된 ERSVP-라우터는 \( R^C \) 값을 사용하는, 즉 기존 RSVP 절차와 역방향 호환성을 유지할 수 있다. 참고로 기존 RESV 메시지 왕복, 즉 Length 값이 10일 경우 ERSVP-라우터는 \( R^C \)값 대신 R값을 사용한다.

RESV 메시지 처리 과정은 전체적인 RSVP 절차 측면에서 기존의 과정과 동일하나 각 ERSVP-라우터 내부의 동작은 큰 차이가 있다. 기존 RSVP 라우터는 Rspect 부분에 실려오는 R 값의 역 약속을 자원 예약을 요청한 호출을 위해 할당하도록 WFQ 기반 스케줄러에게 요구하는 반면 ERSVP-라우터는 LOFQ 스케줄러에게 호출의 지연규약과 요구속도를 충족하는 최적의 자원할당을 요구하기 때문이다. 그림 3에 도시된 RESV 메시지 수신시 ERSVP-라우터의 동작을 구체적으로 살펴보자. RESV 메시지를 받은 ERSVP-라우터는 Rspect에 담겨있는 \( R^C \)값과 Tspect에 담겨있는 M값으로 자신의 약속에만 해당되는 지연규약 b를 다음 식으로 계산한다: \( b = C/R^C + D_1 \). 그런 후 Tspect에서 담겨있는 요구속도 r과 계산한 지연규약 b 정보로 LOFQ 스케줄러에게 호출수락을 요청하고, LOFQ 스케줄러의 호출수락이 성공하면 자원 예약으로 전달한다.

그림 3. PATH 메시지 처리도

그림 4. 보안형 서비스에 대한 FLOWSPEC Object

**그림 2. ERSVP-라우터 구성도**

```plaintext
RSVP 메시지

라우팅

RSVP 메시지

장거리

패킷

패킷분류

패킷

들에서 일부 스케줄러 총속 부분이 수정된다. 특히 RSVP 모듈에 스케줄러 적용부 모듈을 추가하여 RSVP 모듈을 패킷 스케줄러와 분리함으로써 RSVP 모듈의 독립성을 확보한다.

ERSVP-라우터의 RSVP 절차와 자원할당 과정을 살펴보자. RSVP 절차 중 PATH 메시지를 생성하여 호출의 중간간 경로를 통해 전달하고 경로상의 각 라우터가 PATH 메시지를 처리하는 과정은 WFQ 스케줄러를 기반으로 하는 기존의 RSVP 경우와 동일하다. 다만 그림 3과 같이 Adspect는 C와 D 값에 자신의 값과 D 값을 더할 때 LOFQ 스케줄러에 대한 C와 D 값이 아니라 WFQ 기반 스케줄러에 대한 값을 사용한다. 각 라우터에서 호출에 할당하는 스케줄링 속도는 바로 중간간 예약속도 R이므로 (2.3)과 (3.1)로부터 C는 호출의 최대 패킷크기 M인 되고 D는 \( \frac{M^{\text{max}}}{C} \)가 된다.

PATH 메시지를 전달받은 채석측은 (3.2)에 따라 중간간 임계 예약속도 \( R^C \)과 중간간 예약속도 R를 모두 계산한 후 이를 RESV 메시지의 FLOWSPEC 객체에 담아 호출의 역 경로를 통해 경로상의 각 라우터에게 알려준다. 이를 위해 그림 4와 같이 R과 속도 S 영역만 갖는 기존 Rspec을 R, S 및 \( R^C \)의 세 영역을 갖도록 확장한다. RESV 메시지의 FLOWSPEC 객체에 첨부된 Rspec을 담은 방법은 그림 4와 같다. \( R^C \)을 추가하기 위해 Length 영역을 기존 10에서 11로 증가시키고 12번째 워드(Word)에 \( R^C \)값을 수록한다. 이렇게 함으로서 기존 RSVP-라우터는 R 값을 사용하고 제안된 ERSVP-라우터는 \( R^C \) 값을 사용하는, 즉 기존 RSVP 절차와 역방향 호환성을 유지할 수 있다. 참고로 기존 RESV 메시지 왕복, 즉 Length 값이 10일 경우 ERSVP-라우터는 \( R^C \)값 대신 R값을 사용한다.

RESV 메시지 처리 과정은 전체적인 RSVP 절차 측면에서 기존의 과정과 동일하나 각 ERSVP-라우터 내부의 동작은 큰 차이가 있다. 기존 RSVP 라우터는 Rspect 부분에 실려오는 R 값의 역 약속을 자원 예약을 요청한 호출을 위해 할당하도록 WFQ 기반 스케줄러에게 요구하는 반면 ERSVP-라우터는 LOFQ 스케줄러에게 호출의 지연규약과 요구속도를 충족하는 최적의 자원할당을 요구하기 때문이다. 그림 3에 도시된 RESV 메시지 수신시 ERSVP-라우터의 동작을 구체적으로 살펴보자. RESV 메시지를 받은 ERSVP-라우터는 Rspect에 담겨있는 \( R^C \)값과 Tspect에 담겨있는 M값으로 자신의 약속에만 해당되는 지연규약 b를 다음 식으로 계산한다: \( b = C/R^C + D_1 \). 그런 후 Tspect에 담겨있는 요구속도 r과 계산한 지연규약 b 정보로 LOFQ 스케줄러에게 호출수락을 요청하고, LOFQ 스케줄러의 호출수락이 성공하면 자원 예약으로 전달한다.

**그림 2. ERSVP-라우터 구성도**

```
```
의 성공을 통보한다.


4. LOFQ 스케줄러의 성능 특성

5장에서 ERSVP-라우터의 성능을 평가하기 앞서 LOFQ 스케줄러에서 어떻게 대역폭 이용 효율이 개선되는지 살펴보고, 이로부터 대역폭 이용도 측면에서의 몇 가지 성질을 도출한다. 먼저 호름의 그의 요구 속도 r과 지역규격 b의 관계 측면에서 분류한다. (2.2)에 따르면 b는 r의 반비례하므로 r과 b 대신 r과 r의 관계로 환율을 분류하면 정의 1과 같다.

정의 1: 호름의 유형을 다음과 같이 정의한다.
1) 속도제약(RR: Rate-Restricted) 호름: \( r > r_c \), 즉 속도가 지연보다 더 엄격한 환율로 지역규격과 고유 레이턴스의 차이만큼의 지연시간에 여유가 있는 호름입니다.
2) 지역제약(DR: Delay-Restricted) 호름: \( r \leq r_c \), 즉 지연이 초과보다 더 엄격하여 지역규격과 고유 레이턴스 만큼 지연시간이 부족한 호름입니다.
3) 최적(OPT: Optimal) 호름: \( r = r_c \)인 호름입니다.

LOFQ 스케줄러의 대역폭 이용 효율 개선을 이해하기 위해 General-time에 기초한 타임스탬프 계산식 (2.7)을 살펴보자. (0.1) 범위내의 \( \beta \)를 갖는 임의의 호름 \( H \)과 WFQ 방식(\( \beta = 1 \)인 특수한 경우)에 비해 \( \Delta T = (1-\beta)M/H \) 만큼 타임스탬프 값의 줄임이 가능하다. 따라서 \( \Delta T \)만큼 피드백을 빨리 전달하므로 레이턴스 \( \Delta T \)만큼 줄일 수 있다. 여기서 \( M \)는 호름 \( H \)의 최대 피크크기가다. 하지만 나머지 모든 호름들에게 \( \Delta T \)의 추가지연이, 즉 레이턴스 바이어스를 초래한다. [16]. 그러므로 RR 호름의 경우 여유 지연시간으로 추가지연을 상쇄할 수 있어 지역규격의 준수에 문제가 없다. 결과적으로 DR 호름의 경우 예약속도를 높이지 않아도 자신의 레이턴스를 줄일 수 있어 대역폭 이용 효율을 개선할 수 있게 된다. 정리하자면 WFQ 방식에서 RR 호름의 여유 시간을 놓고 있지 않지만 LOFQ 방식에서는 이를 지연자원에 측정하고 DR 호름이 과잉예약 대신 촉진된 지연자원을 재활용하여 성능개선을 이루는 것이다. 이러한 성능 개선 특성으로 인해 RR 호름의 속도제약 정도, DR 호름의 지역제약 정도 그리고 두 호름 유형의 비율이 대역폭 이용도의 개선 정도에 큰 영향을 미치게 된다.

이제 대역폭 이용도 측면에서의 LOFQ 스케줄러의 몇 가지 성질을 도출한다. 호름의 최대 피크크기 M와 지역규격 b의 변화에 따른 대역폭 이용도 특성을 살펴보자. WFQ의 경우 (2.3)로부터 M이 증가할수록 그리고 b가 엄격할수록 레이턴스 q가 지역규격 b를 초과하지 않도록 하기 위해 스케줄링 속도 s를 증가시켜야 하므로 그만큼 대역폭 이용도가 증가한다. GFQ의 경우 (2.4)로부터 M이 증가할수록 그리고 b가 엄격할수록 스케줄링 속도 s를 증가시켜야 하므로 그만큼 대역폭 이용도가 증가한다. 이로부터 다음의 성질(property) 1과 성질 2가 도출된다.

성질 1: WFQ와 GFQ 모두 호름의 최대 피크크기가 개질수록 대역폭 이용도가 낮아지고 반대로 최대 피크크기가 작아질수록 대역폭 이용도가 좋아진다.

성질 2: WFQ와 GFQ 모두 호름의 지역규격이 엄격할수록 대역폭 이용도가 낮아지고 반대로 호름의 지역규격이 느슨할수록 대역폭 이용도가 좋아진다.

(2.2)로부터 M이 증가할수록 그리고 b가 엄격할수록, 즉 줄여들수록 임계 요구속도 \( r^* \)가 높아진다. 따라서 정의 1에 의해 다음의 성질 3과 4가 도출된다.

성질 3: 호름의 최대 피크크기가 케릴수록 RR 호름의 이용은 줄어들과 DR 호름의 이용은 늘어난다. 반대로 호름의 최대 피크크기가 작아질수록 RR 호름의 이용은 늘어나고 DR 호름의 이용은 줄어든다.

성질 4: 호름의 지역규격이 엄격할수록 RR 호름의 이용은 줄어들고 DR 호름의 이용은 늘어난다. 반대로 호름의 지역규격이 느슨할수록 RR 호름의 이용은 늘어나고 DR 호름의 이용은 줄어든다.

호름의 지역규격이 평균분포를 따르 경우 이의 표준편차에 따른 특성을 살펴보자. 표준편차가 증가할수록 엄격한 지역규격과 느슨한 지역규격의 호름이 상대적으로 증가한다. 따라서 보다 속도가 더 심하게 제약되는 호름, 이하 RR 호름과, 보다 지역이 더 심하게 제약되는 호름, 이하 DR 호름의 비율이 높아진다. 따라서 다음의 성질 5가 도출된다.

성질 5: 호름 지역규격의 표준편차가 증가할수록 RR 호름과 DR 호름의 비율이 증가한다.
5. 성능평가

평가 네트워크를 그림 5와 같이 구성하였다. 구체적으로 실험에서는 12개의 노드로 구성되고 흐름 연결 측면에서 노드 0, 1, 2, 3, 4 및 9를 지나가는 $a_1 \rightarrow a_2$ 루트, 노드 5, 12, 2, 3, 7 및 11을 지나가는 $b_1 \rightarrow b_2$ 루트 및 노드 6, 2, 3, 7, 8 및 10을 지나가는 $c_1 \rightarrow c_2$ 루트의 3개 루트를 갖는다. 12개의 노드 중 노드 9, 10 및 11에서 단단 $a_2$, $b_2$ 및 $c_2$로 연결되는 화면은 출력링크가 아니라 착신단말로 연결되는 일종의 가입자 화면으로서 QoS 라우팅 기능을 적용하지 않는다. 따라서 세 개의 루트 모두 5개의 QoS 라우터를 거치며 RSVP 정책에 의해 자원을 예약한다. 모든 링크의 링크지연은 1ms, 모든 착신노드의 지연은 1ms로 가정한다. 각 링크는 모두 2Gbps의 용량을 갖는다.


$$f_{X|x>a\&x<b}(x) = \frac{f_X(x)}{P[a \leq X \leq b]} \text{여기서 } f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$ (5.1)

(5.1)에서 $X$의 기대값을 의미하는 $\mu$는 $(a+(b-a)/2)$의 값을 갖고, $\sigma$는 $X$의 표준편차이다.

다음으로 평가에 사용한 호름의 요구속도에 대해 살펴본다. 음성전화나 영상통화 등이 대부분 고정속도 트래픽으로 모형화되므로 본 연구에서는 고정속도 호름만 취급한다. 호름 요구속도 측면에서 8Kbps의 음성, 128Kbps의 오디오 및 1Mbps의 영상 호름의 7:2:1의 비율로 랜덤하게 발생하는 VoIP(Voice over IP) 작업부하를 고려한다.


먼저 균등분포 지연규격에 대해 예를 방식의 성능을 평가한다. 패킷 크기 증가에 따른 수락률을 수와 수락률의 수 속도에서 RSVP-라우터 대비 ERSVP-라우터의 이득을 균일과 7과 8에 각각 도시하였다. 3개의 루트가 수락한 전체 호름의 수를 도시한 그림 6을 고찰해보자. 패킷 크기가 증가함수록 수락률 수가 줄어드는 현상은 성질 1에 의해 대역폭이

![그림 5. 성능평가 네트워크](image)

![그림 6. 수락률 수 비교(균등분포 지연규격)](image)
용도가 줄어들기 때문이다. 이론적미터 이론적미터서 낮은지연으로 염격해질수록 수락 호름 수가 줄어드는 것은 성절 2에 따르 것으로 이해할 수 있다. RSVP-라우터 대비 ERSVP-라우터가 더 많은 수락 을 수락하는 비율을 도시할 그림 7을 살펴보자. 패킷 크기가 증가함수록 이드가 사라지게 감소함을 볼 수 있다. 이는 성절 3에 따른 RR 호름과 DR 호름의 비율 변화로 설명할 수 있다. 두 호름의 비율이 비슷한 영역에서는 RR 호름의 낮비 자연자원을 충분히 제한적용 할 수 있어 이드가 증가 반면 패킷크기가 큰 영역에서는 DR 호름의 비율이 더 높아지어서 이드가 다소 줄어들 수 있다. 그림 7은 또한 자연적임이 복잡지연과 낮은지연인 경우 최대 30%의 이득을 얻을 수 있으나 낮은지연인 경우 이드가 7~8%로 낮아짐을 보여준다. 자연적임과 낮은지연의 경우 RR 호름과 DR 호름의 비율이 수락하여 자연자원의 생산과 제한적용이 잘되기 때문에 높은 이득을 보이나, 낮은지연의 경우 성절 4로부터 DR 호름의 비율이 낮아져 과도적인 적게 일어나고 그 결과 LOFQ에 의한 성능 개선의 여지가 그 만큼 줄어들기 때문에로 이해할 수 있다. 

다음은 정규분포 자연적임에 대해 성능을 평가한 다. 여러 알려진 바와 같이 정규분포 자연적임에 의해 각 라우터에 배분되는 자연적임의 분포는 정규분포로 수렴하므로 균등분포 자연적임 보다 실제적인 성능을 관할할 수 있다. 2Kbit의 패킷크기에서 표준편차의 증가에 따른 수락 호름 수의 수락호름의 수 측면에서 RSVP-라우터 대비 ERSVP-라우터의 이득은 그림 8과 9에 각각 도시하였다. 그림 8에서 자연적임너의 전달지연과 낮은지연의 경우 표준편차가 증가할 때 RSVP-라우터의 경우 수락 호름 수가 감소하지만 ERSVP-라우터의 경우 호름수락 능력을 계승 유지함을 관찰할 수 있다. 이 결과 그림 9에서 관찰할 수 있도록 표준편차가 증가할수록 ERSVP-라우터의 이득이 증가한다. 이러한 특성을 사항적 5에 따른 DR 호름과 RR 호름의 비율증가로 이해할 수 있다. WFQ하에서는 경우 DR 호름 수의 증가에 따라 과다예약이 생겨서 대역폭 이용능도가 크게 하되 반면 WFQ하에서는 RR 호름의 증가에 의해 다가 발생하는 자연자원을 정도 DR 호름이 더 많이 제한적용하기 때문에 높은 대역폭 이용효율을 유지함을 보여준다.
RSVP 모듈과 수락세어 모듈에서 일부 스케줄러 종속 부분을 수정하였다. 그리고 기존의 RSVP 경사와 메시지를 그대로 사용하나 종단간 예약속도 R 및 슬래 S와 더불어 종단간 지연치 가측에서 종단간 최적 예약속도를 의미하는 임계 예약속도 R^c를 경로상의 모든 라우터에 통지하기 위해 RESV 메시지의 FLOWSPEC 객체내 Rspec을 확장하였다. 구체적으로 FLOWSPEC 객체의 Length 영역을 기존 10에서 11로 증가시키고 12번째 워드(word)에 R^c 값을 수록하였다. 기존의 RSVP-라우터는 R 값을 사용하고 제안된 RSVP 라우터는 Length 영역 값을 확인하여 10의 값을 가질 경우는 R값을 사용하고, 11의 값을 가질 경우 R^c 값을 사용하도록 함으로서 기존 RSVP와 역방향 호환성을 유지할 수 있었다.

대역폭 이용 효율 측면에서 LOFQ 스케줄러의 성질을 분석한 후 12개의 노드와 3개의 튜브로 구성되는 패기 네트워크에서 전형적인 품질 보장형 서비스인 VoIP 트래픽을 대상으로 시뮬레이션을 통해 ERSVP-라우터의 호흡 수락 능력을 기존 RSVP-라우터의 것과 비교 평가하였다. 평가 결과 기존 RSVP-라우터에 비해 최대 30%까지 더 많은 환점을 수락할 수 있음을 확인하였다.

참고 문헌

[17] 감태준, "최적 레이턴시 기반 공정 큐의 알고리

김태준
1980년 2월 경북대학교 전자공학 과 졸업
1982년 2월 한국과학기술원 전자공학 석사
1999년 8월 한국과학기술원 전자공학 박사
1992년 3월 한국전자통신연구원
1996년 3월 친안공업대학
2005년 3월 ~ 현재 공주대학교 정보통신공학부 교수
관심분야: 고속통신망, VoIP, 트래픽제어