

## RTP 기반의 TCP 친화적인 전송률 조절 기법

이선현<sup>0</sup>, 정광수

광운대학교 전자공학부

sunlee@cclab.kw.ac.kr, kchung@daisy.kw.ac.kr

### TCP-Friendly Rate Control Scheme Based on the RTP

Sunhun Lee<sup>0</sup>, Kwangsue Chung

School of Electronics Engineering, Kwangwoon University

#### 요 약

최근 오디오나 비디오 스트리밍과 같은 멀티미디어 트래픽이 증가하고 있다. 이러한 트래픽들은 패킷을 전달하는데 대부분 UDP(User Datagram Protocol)기반의 RTP(Realtime Transport Protocol)를 사용한다. 하지만 UDP기반의 RTP는 기본적으로 혼잡제어 메커니즘이 없으며 현재 인터넷의 주요 트래픽인 TCP(Transmission Control Protocol)와의 형평성을 보장하지 않는다는 문제점을 갖는다. 본 논문에서는 스트리밍 트래픽의 TCP 친화적인 전송률 조절 기법으로 TF-RTP(TCP-Friendly RTP)를 제안하였다. TF-RTP는 네트워크 상태가 혼잡하여 패킷 손실이 발생할 경우, 개선된 파라미터들을 사용하여 경쟁하는 TCP의 전송률을 보다 정확하게 계산하여 스트리밍 트래픽의 전송률을 조절함으로써 경쟁하는 TCP 트래픽과 친화적으로 동작하며 네트워크 대역폭을 보다 공평하게 사용하게 된다. 실험을 통해 제안한 TF-RTP가 TCP의 전송률을 보다 정확하게 계산하며 TCP 친화성, 공평성 측면에서도 성능 개선을 보임을 확인할 수 있었다.

#### 1. 서 론

현재 인터넷에서의 주요 트래픽들은 전송 프로토콜로 대부분 TCP를 사용하고 있다. TCP는 종단간(end-to-end)의 신뢰적인 패킷 전달을 수행하며 혼잡 제어 메커니즘(congestion control mechanism)을 통해 송신단의 전송률을 직접 제어하여 혼잡 상황으로 발생하는 데이터의 무분별한 손실을 막고 네트워크가 안정하게 동작할 수 있도록 한다 [1].

최근들어 오디오나 비디오 스트리밍과 같은 멀티미디어 트래픽이 증가하고 있다. 이러한 트래픽들은 지연에 민감하고 어느 정도의 패킷 손실을 허용하며 전송률을 기반으로 동작하는 특징을 가진다. 멀티미디어 트래픽의 이러한 본질적인 특징으로 인해 신뢰적인 패킷 전달을 수행하는 TCP는 스트리밍 서비스의 전송 프로토콜로는 적합하지 않다. 따라서 대부분의 멀티미디어 트래픽은 혼잡 제어를 수행하지 않는 UDP를 사용하게 되었다. 하지만 UDP는 TCP와 같은 혼잡 제어 메커니즘이 없으므로 네트워크 혼잡 상황을 야기하여 네트워크를 붕괴(congestion collapse)시킬 가능성이 있으며 기존의 주요 트래픽인 TCP와도 친화적(TCP-friendly)으로 동작하지 않는다는 문제점을 갖는다. 이러한 문제점을 극복하기 위해 1990년대 이후로 스트리밍 전송 기법과 관련하여 네트워크 안정성을 향상시키면서 기존 인터넷의 주요 트래픽인 TCP와의 친화성을 고려한 혼잡 제어 메커니즘에 대한 연구가 활발하게 진행되고 있다.

Padhye는 TCP의 전송률을 RTT(Round Trip Time)와 RTO(Retransmission Time Out), 패킷손실률에 관한 공식으로 모델링하였다 [2]. 이러한 TCP의 전송률 모델링 공식을 표준 스트리밍 전송 프로토콜인 RTP에 적용하여 TCP 친화적이며 TCP와 공평하게 전송률을 조절하는 스트리밍 프로토콜로 S-RTP가 있다 [3]. S-RTP는 네트워크 혼잡에 의한 패킷 손실이 발생할 경우, TCP의 전송률 모델링 공식을 이용해서 전송

률을 조절하는 스트림 전송 기법으로 TCP 친화성을 만족시키고 동시에 전송률의 변화를 줄이는 것을 목적으로 갖는다. 하지만 TCP 전송률 공식에 의해 TCP 친화적인 전송률을 구하는 과정에서 부정확한 파라미터 값들을 사용하기 때문에 TCP의 전송률을 정확하게 계산할 수 없으며 이러한 문제점을 보완하기 위해 추가적인 알고리즘을 사용하게 된다.

본 논문에서는 멀티미디어 스트림을 전달하는 RTP의 전송률을 TCP 친화적이며 공평한 전송률로 조절하는 전송 기법으로 TF-RTP를 제안하였다. TF-RTP에서는 RTT, RTO, 패킷손실률의 파라미터들을 사용하여 경쟁하는 TCP의 전송률을 보다 정확하게 계산한다. 경쟁하는 TCP의 전송률에 대한 정확한 계산을 바탕으로 TF-RTP는 추가적인 알고리즘의 적용없이 네트워크 상태에 따라 스트리밍 트래픽의 전송률을 적절하게 조절하며 이를 통해 전송률의 변화를 줄이고, 멀티미디어 스트림의 TCP 친화성 및 공평성을 크게 개선하였다.

논문의 2장에서는 TCP 친화성과 공평성을 고려한 기존 RTP 기반의 전송 프로토콜에 대한 소개와 함께 문제점을 기술하였으며 3장에서는 제안한 TF-RTP의 세부 알고리즘을 소개하였다. 4장에서는 실험을 통해 제안한 알고리즘의 성능을 평가하였으며 마지막으로 5장에서는 결론을 맺었다.

#### 2. 관련 연구

##### 2.1 TCP 전송률 모델링 공식

Padhye는 [2]에서 TCP Reno를 기반으로 RTT와 RTO, 패킷손실률과 같은 파라미터들을 사용하여 TCP 플로우의 전송률에 대한 공식을 모델링하였다.

$$T = \frac{S}{t_{RTT}\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1+32p^2)} \quad (1)$$

\* 본 논문은 유비쿼터스 컴퓨터 프론티어 사업단의 지원에 의한 연구결과입니다.

식 (1)은 Padhye의 TCP 전송률 모델링 공식을 보이고 있다.  $t_{RTT}$ 는 RTT를,  $S$ 는 패킷의 크기를 의미하며,  $p$ 는 패킷손실

를,  $t_{RTO}$ 는 RTO를 의미한다. Padhye의 TCP 전송률 모델링 공식은 동일한 네트워크 상태에서 TCP의 평균 전송률을 계산할 수 있다는 장점으로 인해 멀티미디어 스트림의 전송률을 TCP 친화적이며 공평하게 조절하고자 하는 스트리밍 프로토콜에 관한 기존 연구들에서 빈번히 사용되고 있다.

2.2 S-RTP 기법

S-RTP는 표준 스트리밍 프로토콜인 RTP를 네트워크 혼잡 상태를 회피하고 전송률의 변화를 줄이며 인터넷의 주요 트래픽인 TCP와 친화적으로 동작할 수 있도록 전송률 조절 기법을 적용한 스트리밍 전송 기법이다. S-RTP는 RTP의 제어 프로토콜인 RTCP의 정보를 이용하여 RTT, RTO, 패킷손실률의 파라미터들을 계산하고 이를 Padhye의 TCP 전송률 모델링 공식에 적용하여 TCP와 친화적이며 공평한 전송률을 계산한다.

$$p = \frac{N_{real}}{(N_{max} - N_{first})} \quad (2)$$

$$t_{SRTT} = (\alpha \times t_{currentRTT}) + (1 - \alpha) \times t_{SRTT} \quad (3)$$

식 (2)는 S-RTP에서 사용하는 패킷손실률 계산식을 보이는 것으로  $N_{real}$ 은 수신된 전체 패킷의 수를,  $N_{max}$ 은 수신된 패킷의 최대 시퀀스 번호를,  $N_{first}$ 은 최초에 수신한 패킷의 시퀀스 번호를 의미한다 [4]. 식 (3)은 SRTT(Smoothed RTT)를 구하는 식으로 TCP와 유사하게  $\alpha$ 를 0.8로 설정하게 된다.

패킷 손실이 발생할 경우, S-RTP는 식 (2), (3)을 통해 구해진 파라미터들을 TCP 전송률 모델링 공식에 적용하여 TCP 친화적이며 공평한 전송률을 계산한다. 계산된 전송률은 네트워크 혼잡 상태 회피와 TCP 친화성 및 공평성을 고려하여 그림 1과 같은 방법에 의해 플로우의 전송률을 조절하게 된다. 패킷 손실이 발생하지 않은 안정한 네트워크 상태로 판단될 경우에는 TCP와 유사하게 전송률을 증가시킨다.

IF,  $p > 0$  : Congestion State

IF,  $(T_{S-RTP} > T_{TCP})$

New  $T_{S-RTP} =$

$$MAX(\beta \times T_{S-RTP} + (1 - \beta)(T_{S-RTP}(1 - \sqrt{p})), T_{TCP})$$

그림 1. S-RTP의 전송률 조절 기법

S-RTP는 패킷 손실이 발생하여 네트워크가 혼잡상태로 판단되면 네트워크 상태를 안정화하고 TCP 친화적인 방법으로 전송률을 조절한다. 하지만 Padhye의 공식에 사용되는 파라미터들의 부정확성때문에 경쟁하는 TCP 플로우의 전송률을 정확하게 계산할 수 없으며 이로인해 실제 전송률을 조절할 경우, TCP 전송률 공식에 의해 계산된 전송률이 아닌 그림 1과 같은 추가적인 알고리즘에 의해 결정된 전송률을 사용하게 된다. 이러한 문제는 TCP의 전송률을 구하는데 사용한 파라미터들의 부정확성에 의해 발생하는 문제이다.

3. TF-RTP 알고리즘

3.1 정확한 파라미터 산출

본 논문에서는 스트리밍 트래픽의 전송률을 TCP와 친화적이며 공평한 전송률로 조절하는 기법으로 TF-RTP를 제안하였다. TF-RTP는 패킷 손실에 의해 네트워크 상태가 혼잡하다고 판단할 경우, TCP 전송률 공식을 이용하여 스트리밍 트래픽의 전송률을 조절하게 된다. 경쟁하는 TCP 플로우의 전송률을 구하는 방법은 식 (1)을 사용하며 공식에 사용되는 파라미터들의 정확한 값을 얻기 위해 다음의 방법을 사용한다.

- ii) 전송률 변화를 줄이기 위한 Smoothed RTT 계산
- iii) TCP와 유사한 RTO 계산

S-RTP에서 사용하는 RTCP의 패킷손실률 정보를 패킷수 기반의 패킷손실률 정보를 제공한다. 패킷수 기반의 패킷손실률,  $p$ 는 지나치게 큰 값으로 계산되며 실제 TCP 트래픽의 전송률 보다 낮은 값으로 전송률을 계산하게 된다. TF-RTP에서 패킷손실률,  $p$ 는 이러한 문제를 극복하기 위해 식 (4)와 같이 패킷수 기반이 아닌 패킷 손실 이벤트 기반으로 계산된다. 패킷 손실 이벤트 기반의 패킷손실률 계산은 한개 이상의 패킷 손실이 발생하더라도 하나의 패킷 손실 이벤트로 간주하므로 패킷수 기반의 방법에 비해 패킷손실률의 변화가 적으며, 따라서 TCP 전송률 공식에서의 값의 변화를 줄일 수 있다.

$$p = \frac{\text{loss event}}{\#of\ received\ packets/\#of\ loss\ event} \quad (4)$$

TF-RTP에서 RTT와 RTO 파라미터는 TCP와 유사한 방법을 적용하여 TCP의 전송률을 보다 정확하게 계산하게 된다. RTT의 경우, 변화를 줄이기 위해 식 (3)과 같은 저대역 필터링을 통해 1차 SRTT를 계산하여 TF-RTP의 송신단에 알려주며, 송신단에서는 RTCP에 의해 보고된 SRTT 값을 다시 저대역 필터링을 통해 2차 SRTT를 계산한다. RTO의 경우, 식 (5)과 같이 TCP에서와 같은 방법으로 계산한다.

$$t_{RTO} = t_{SRTT} + 4 \times SRTT_{Variation} \quad (5)$$

3.2 TF-RTP의 전송률 조절 알고리즘

TF-RTP는 RTCP 패킷의 패킷손실률 정보를 이용하여 네트워크 상태를 판단하게 된다. 즉, 패킷손실률이 0보다 크면 네트워크를 혼잡 상태로, 0과 같으면 안정한 상태로 판단하게 된다. TCP의 전송률 조절 기법인 AIMD(Additive Increase Multiplicative Decrease) 기법은 패킷 손실이 발생하지 않으면 RTT당 하나의 패킷만큼 전송률을 선형적으로 증가시킨다.

TF-RTP는 TCP의 선형적인 전송률 증가 방법과 유사하게 패킷손실률이 0인 경우, 네트워크 상태를 안정한 상태로 판단하고 기존의 TCP와 유사하게 전송률을 RTT마다 하나의 패킷만큼 증가시켜 나간다. 패킷손실률이 0보다 큰 값을 가질 경우, 네트워크 상태는 혼잡한 상태로 판단된다. 이러한 경우에 TF-RTP는 3.1에서 정리한 방법에 의해 패킷 손실 이벤트에 기반한 패킷손실률을 구하고, TCP와 유사한 방법으로 RTT와 RTO 파라미터들을 계산한다. 이러한 파라미터들을 TCP 전송률 공식에 적용하여 보다 정확하게 TCP 친화적이며 네트워크의 대역폭을 공평하게 점유할 수 있도록 전송률을 계산하게 된다.

그림 2는 TF-RTP의 전송률 조절 알고리즘을 보여준다.  $T_{New}$ 는 3.1에서 기술한 방법에 의해 파라미터들을 계산하여 이를 식 (1)에 대입해서 얻은 전송률을 의미한다. 네트워크 상태가 안정할 경우, TCP와 마찬가지로 RTT당 하나의 패킷만큼 전송률을 증가시키게 된다. TF-RTP에서 전송률을 조절하는 주기는 RTT가 아닌 RTCP 주기이므로 증가되는 패킷의 수는 RTCP 주기를 RTT로 나눈 값으로 계산된다.

If ( $p > 0$ ) : Congestion State

Sending Rate =  $T_{New}$

Else ( $p == 0$ ) : Stable State

Increase Packet =  $\frac{RTCP\ period}{t_{SRTT}}$

Sending Rate = Current Rate + Increase Packet

그림 2. TF-RTP 전송률 조절 알고리즘

- i) 패킷 손실 이벤트 기반의 패킷손실률 계산

4. 실험 및 성능 평가

4.1 실험 환경

제안한 TF-RTP의 성능 평가를 위해 LBNL(Lawrence Berkely National Laboratory)의 ns-2(network simulator)를 사용하여 실험을 수행하였다 [5]. 그림 3은 TF-RTP의 성능 평가를 위한 실험 환경을 보여준다. TF-RTP가 계산하는 TCP 전송률의 정확성을 검증하기 위해 경쟁하는 트래픽으로 TCP를 사용하였으며 성능 평가는 TF-RTP와 TCP 트래픽의 전송률 변화 및 평균 전송률 비교를 통해 분석하였다.

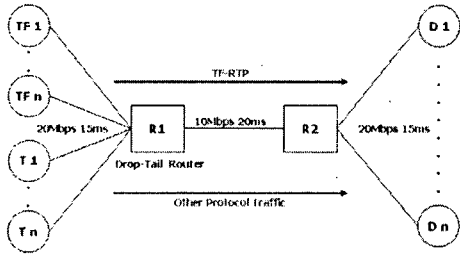


그림 3. 실험 환경

4.2 TF-RTP의 성능 검증

그림 4는 TF-RTP에서 사용하는 새로운 파라미터들을 적용한 TCP 전송률 계산의 정확성을 검증하는 실험 결과이다. 두 개의 TCP 전송률은 그림 3의 실험 환경에서 TF-RTP 트래픽과 경쟁하는 TCP 트래픽의 전송률 변화를 보여주고 있다. TF-RTP의 전송률  $T$ 는 새로운 파라미터들을 사용해서 계산한 전송률을 나타내며 S-RTP의  $T$ 는 이전 파라미터들을 사용해서 계산한 결과를 나타낸다. 그림 4에서 두 TCP의 평균 전송률은 3.11Mbps로 측정되며 S-RTP의  $T$ 는 평균 1.68Mbps로 TCP 트래픽의 평균 전송률보다 낮은 값으로 계산된다. 이러한 결과는 S-RTP에서  $T$ 의 계산에 사용되는 RTT, RTO, 패킷손실률 파라미터들의 부정확성에 근거한다. 하지만 TF-RTP의  $T$ 는 평균 3.03Mbps로 TCP의 평균 전송률과 비슷하게 계산된다. 그림 4의 실험 결과를 통해 TF-RTP가 가지는 새로운  $T$ 의 계산이 경쟁하는 TCP 플로우의 평균 전송률을 거의 정확하게 계산할 수 있음을 확인할 수 있다.

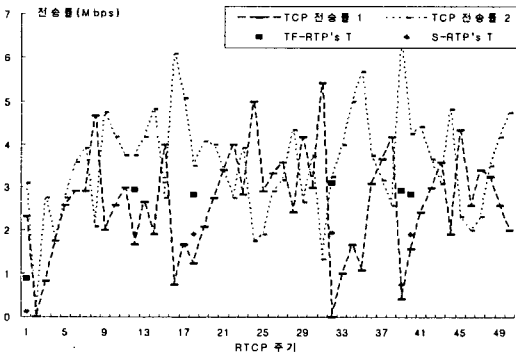


그림 4. TCP 전송률 계산

그림 5는 3.2절에서 기술한 TF-RTP의 전송률 조절 알고리즘의 성능을 검증한 실험 결과이다. 실험 결과는 경쟁하는 두 개의 TCP 트래픽과 TF-RTP 트래픽의 전송률 변화를 보여준다. 결과에서, TCP 트래픽의 평균 전송률은 3.2Mbps이며 TF-RTP의 평균 전송률은 3.1Mbps로 산출된다. 그림 4와 그림 5의 실험 결과를 통해 TF-RTP는 경쟁하는 TCP와 친화적으

로 전송률을 조절하며 네트워크의 대역폭을 공평하게 사용하는 것을 확인할 수 있다. 또한 네트워크 상태 혼잡에 따른 전송률 변화가 감소하여 안정한 전송률을 유지할 수 있다.

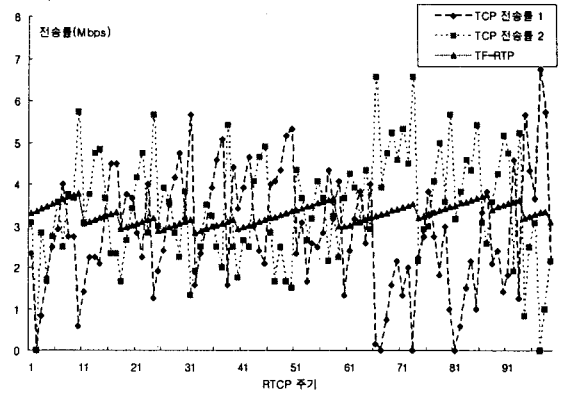


그림 5. 전송률 변화

5. 결론

멀티미디어 스트리밍에서 패킷 전달은 대부분 UDP기반의 RTP를 통해 이루어지고 있다. 이러한 UDP기반의 RTP는 기본적으로 혼잡제어 메커니즘이 없으며 현재 인터넷의 주요 트래픽인 TCP와의 형평성을 보장하지 않는다는 문제점을 갖는다. 이러한 문제를 해결하기 위해 TCP 친화적으로 전송률을 조절하는 S-RTP가 제안되었으나 TCP의 전송률을 정확히 계산하지 못한다는 한계를 가지고 있다.

본 논문에서는 이러한 S-RTP의 가장 큰 문제점인 TCP 전송률 계산의 부정확성을 개선하기 위해 수정된 파라미터 계산 방법을 제안하였으며 이를 통해 보다 정확하게 TCP의 전송률을 계산하여 스트림의 전송률을 조절하는 TF-RTP를 제안하였다. S-RTP가 TCP 친화적이고 안정적인 전송률 변화를 위해 추가적인 알고리즘을 사용하는 반면, TF-RTP는 정확한 TCP 전송률 계산을 바탕으로 추가적인 알고리즘없이 TCP 친화적이고 공평한 전송률 조절이 가능하게 된다. 실험을 통해 TF-RTP의 성능을 분석하였으며 그 결과로 기존의 방법에 비해 TCP 친화성과 네트워크 대역폭 사용에 대한 공평성이 크게 개선됨을 확인하였다.

향후 과제로는 실험 환경을 확장하여 다양한 환경에서 TF-RTP에 대한 성능 측정과 함께, 전송률의 변화를 줄이는 방법에 대한 연구가 수행되어야 할 것이다.

참고 문헌

- [1] S. Floyd and F. Kevin, "Router mechanisms to support end-to-end congestion control," *Technical Report, LBL-Berkeley*, February 1997.
- [2] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," *ACM SIGCOMM*, 1998.
- [3] B. Song, K. Chung, and Y. Shin, "SRTP: TCP-friendly congestion control for multimedia streaming," *16th International Conference on Information Networking*, January 2002.
- [4] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," *IETF, RFC 1889*, January 1996.
- [5] The network simulator ns-2, <http://www.isi.edu/nsnam/ns/>