# 유·무선 네트워크에서 무선 손실률을 고려한 equation 기반의 멀티캐스트 기법

박수현O, 안홍범, 홍진표
한국외국어대학교 컴퓨터 및 정보통신공학과
{parksh, mythopoeic, jphong}@hufs.ac.kr

# The equation-based scheme for multicast considering wireless loss probability in wired and wireless networks

Soohyun ParkO, Hongbeom Ahn and Jinpyo Hong
The department of Information and Commuications Engineering,
Hankuk University of Foreign Studies

**요 약**

TFMCC(TCP-Friendly Multicast Congestion Control)방식은 equation 기반의 멀티캐스트 혼잡 제어 메커니즘으로 TFRC(TCP-Friendly Rate Control) 프로토콜을 유니캐스트에서 멀티캐스트 도메인으로 확장한 방식이다. TFMCC 방식은 현재 무선 환경에 적용 시 유선 환경에서의 혼잡에 의한 패킷 손실뿐만 아니라, 무선 환경에서 무선 링크 에러를 네트워크의 혼잡으로 인식하며, single-rate 멀티캐스트 혼잡제어의 특성인 가장 낮은 수신단의 성능으로 전체 네트워크 전송률이 급격히 저하된다. 이에 본 논문에서는 무선 환경에서의 TFMCC의 성능 향상을 위해 네트워크의 무선 환경의 손실률과 유선 환경 손실률을 모델링하여 구분한 ARC(Analytical Rate Control)의 TCP 전송률 equation 을 TFMCC에 맞게 적용하였으며, 멀티캐스트 도메인에서 전송률 제어 시 무선 손실률을 별도로 고려하는 방식(M-ARC)을 제안하였다. 또한 성능 평가를 위해서 시뮬레이션 한 결과 무선 환경을 고려한 M-ARC(Multicast-Analytical Rate Control)가 기존의 TFMCC에 비해 더 높은 전송률을 유지함을 볼 수 있었다.

## 1. 서 론

혼잡제어 메커니즘은 인터넷의 best-effort 서비스를 함에 있어 중요하게 고려되어야 할 요소이다. 현재 다수의 인터넷 트래픽에서 TCP(Transmission Control Protocol)를 기반으로 한 혼잡제어 메커니즘을 사용하고 있다. 그러나, AIMD(Additive Increase Multiplicative Decrease) 혼잡제어를 사용하는 TCP는 패킷 손실에 대해 전송률을 급격히 저하시켜 실시간 스트리밍 서비스시 오디오나 비디오의 품질을 떨어뜨려 서비스 제공이 어려운 문제점이 있다. 이를 해결하기 위해, UDP를 사용하고 이를 다수 수신자에게 전송할 때 동일 그룹의 수신자에게 멀티캐스트 방식을 이용하여 전체 네트워크 트래픽을 경감시킬 수 있다.

반면에 기존의 유니캐스트 방식의 TCP 트래픽과 멀티캐스트 방식의 UDP(User Datagram Protocol) 트래픽이 공존시 UDP 트래픽은 혼잡제어를 하지 않아 다량의 트래픽을 사용하므로, 다른 프로토콜과의 대역폭 사용에 있어 공정성 문제를 야기한다.

이와 같은 문제점 해결을 위한 TCP-friendly 혼잡제어 기법 중 유니캐스트 방식으로 TFRC(TCP-Friendly Rate Control)[1]가 제안되었다. TFRC는 전송률을 equation 기반으로 제어하며, 이를 통해 네트워크 혼잡 시 대역폭을 조절하는 기능을 수행한다. TFRC는 유니캐스트에 적용 가능한 프로토콜로, 이를 멀티캐스트 도메인에 적용한 메커니즘으로 TFMCC(TCP-Friendly Multicast Congestion Control)[2]가 있다. TFMCC는 PGMCC(PGM Congestion Control)[3]와 함께 IETF[4]의 RMT(Reliable Multicast Transport) WG을 중심으로 진행된 대표적인 멀티캐스트 혼잡제어 기법 중의 하나이며, 단일 전송률을 기반으로 멀티캐스트 수신 그룹 중에서 가장 낮은 전송 속도의 수신자에 송신자의 전송률을 적용시키는 방식이다.

혼잡제어를 하기위해 TFMCC는 기존의 TCP 전송률 equation[5]을 이용하여 TCP 전송률을 제어하고, 송신자로의 피드백을 위한 전송률을 계산한다. Equation 기반의 모델링으로써, 부드러운 전송률의 변화를 통한 실시간 스트리밍 서비스에 장점을 가진다. 하지만 기존의 TFMCC를 비롯한 다수의 연구 논문들은[8],[9],[10] 모두 패킷 손실률이 낮은 유선 환경을 대상으로 설계되었으므로, 이를 무선 환경에 적용시 유선 환경에서의 네트워크 혼잡에 의한 패킷 손실 뿐만아니라 무선 환경에서 발생하는 다수 요인에 의한 링크 에러를 패킷 손실로 인식하여, 전송률이 급격히 저하되는 문제점이 있었다.

이를 보완하기 위한 방법으로 본 논문에서는 기존의

TFMCC 방식에 무선 환경에서의 실시간 트래픽을 위한 전송률 제어 기법[6]을 적용하여, TFMCC를 무선 환경에 효율적으로 동작할 수 있는 개선 방안을 제안하였다.

본 논문의 구성은 다음과 같다. 1장 서론에 이어 2장에서는 관련 연구로써, 멀티캐스트 혼잡제어 기법인 TFMCC와 무선 환경에서의 무선 링크 에러를 다룬 기법인 ARC(Analytical Rate Control)에 대해 알아본다. 3장에서는 무선 환경에서의 TFMCC 동작을 보완하기 위해, 본 논문에서 제안하는 방안인 M-ARC(Multicast-Analytical Rate Control)에 대해 논한다. 4장에서는 제안하는 방안의 성능 평가를 위한 토폴로지와 시나리오 구성 및 성능 측정 결과를 분석하고, 마지막으로 5장에서 결론을 맺는다.

## 2. 관련연구

### 2.1. TFMCC

TFMCC[2]는 TCP-friendly 전송률을 계산하기 위해 TFRC와 다음 식 (1)과 같은 TCP 전송률 equation을 사용하여 수신자 측에서 송신자의 전송률($T_{tcp}$)을 계산하게 된다. 여기서 $t_{RTT}$는 송신자와 수신자 간의 RTT(Round Trip Time) 값이고, p는 패킷 손실률이며, s는 패킷의 크기를 의미한다[5].

$$T_{tcp} = \frac{s}{t_{RTT}\left(\sqrt{\frac{2p}{3}} + \left(12\sqrt{\frac{3p}{8}}\right)p(1+32p^2)\right)} \quad (1)$$

각 수신자는 자신이 참여한 멀티캐스트 그룹의 패킷 손실률과 RTT값을 측정하여 송신자에게 알려주고, 송신자는 (1)에 의해서 모델링한 전송률을 가지고 멀티캐스트 패킷을 송신하게 되는 것이다. TCP-friendly 관점에서는 만약 송신자가 특정 수신자에 대해서 이 전송률을 초과하지 않으면 동일 병목 링크상의 TCP 플로우에 영향을 주지 않는다는 점을 알 수 있다.

멀티캐스트 세션 내에서 혼잡제어를 위해 송신자는 수신자들에게 피드백 메시지를 받는다. 이 때, 수신자가 송신자의 현재 전송률보다 더 낮은 전송률을 가리키는 피드백을 보내면, 송신자는 즉시 피드백 메시지의 전송률로 감소시킬 것이다. 하지만, 수신자는 피드백 폭주현상을 막기 위해서 계산된 전송률이 현재 전송률보다 작지 않으면 피드백 메시지를 보내지 않는다. 이러한 방식은 피드백으로 인한 트래픽이 폭주하지 않는다는 장점이 있는 반면에, 전송률 증가에 있어서 문제가 발생한다.

따라서 [2]는 전송률 증가를 위해 CLR(Current Limiting Receiver)이라는 개념을 사용한다. CLR은 송신자가 멀티캐스트 그룹 내의 가장 낮은 예상 전송률을 가진 것으로 판단한 수신자이다. CLR은 피드백 메시지를 억제하지 않고 즉시 피드백을 전송하도록 허가되었다. 따라서 가장 낮은 처리율을 가진 수신자(CLR)의 네트워크 상황이 좋아진다면 더 높은 전송률을 계산해서 송신자에게 전송하게 되고, 송신자는 이를 즉각 반영하여 전송률을 증가시킬 수 있다.

### 2.2. ARC

ARC[6]는 equation 기반의 메커니즘으로 무선 환경에서 발생할 수 있는 다양한 요인의 패킷 손실률을 고려하여 TCP의 ideal behavior를 모델링했다. [6]은 유니캐스트 도메인 내에서 무선 손실률을 고려한 첫 번째 모델로써, 다음 식 (2)를 사용한다.

$$T = \frac{s}{4 \cdot t_{RTT}} \cdot \left(3 + \sqrt{25 + 24\left(\frac{1-\omega}{\pi-\omega}\right)}\right) \quad (2)$$

T는 packets/sec 단위의 전송률이며, $t_{RTT}$는 round trip time, ω는 무선 링크 에러를 나타내기 위한 패킷 손실률을 나타내고, π는 네트워크 혼잡과 무선 링크 에러를 포함한 전체 패킷 손실률이고, s는 패킷의 크기를 의미한다. ARC의 전송률 equation에서 유선 환경에서 발생하는 네트워크 혼잡에 의한 패킷 손실률을 정리하면, 다음 식 (3)과 같다.

$$p_c = \left(\frac{\pi-\omega}{1-\omega}\right) \quad (3)$$

## 3. M-ARC

ARC[6]의 four-state 마코프 연쇄 모델을 통해, 종단간의 패킷 손실률을 계산했을 시, 식 (3)이 유선 환경에서 TCP 전송률 equation에 적용될 수 있다. 그러나 TFMCC와 ARC의 모델링 환경이 다르므로, 변수에 대한 재정의가 필요하다. [2]의 p와 [6]의 $p_c$, π, ω에 대한 관계를 살펴보면, 다음과 같다.

유선 상황에서 고려된 TCP-friendly 기법들은 무선 상황에 대한 변수가 없기 때문에, 네트워크 혼잡으로 인한 패킷 손실과 무선 환경의 링크 에러에 의한 패킷 손실을 모두 p로 간주한다. 즉, 유선 손실률과 무선 손실률은 독립 사건이 아니기 때문에, 유선 손실과 무선 손실이 동시에 일어나는 경우, 손실률을 계산에 어려움이 있다. 이를 위해, 본 논문에서는 유무선 환경에서 무선 손실률을 고려하여, 이를 유선 손실률에 대해 계산을 하는 식 (3)을 적용하는 것에 초점을 맞추어 기술한다.

ω는 경로 손실, 다중경로 페이딩, 핸드오프로 인한 신호 손실들로 인해 MAC 계층에서 연산되는 무선 링크와 관련된 모든 패킷 손실을 의미한다. 이 때, ω는 무선 링크에서의 패킷 손실률로써, 아래 계층의 MAC 계층에서 정보를 얻는다고 가정하고[6] 앞으로 이를 무선 손실률로 정의한다.

식 (4)에서 결과값은 각각의 손실률을 유선 손실률과 무선 손실률을 포함하여 p에 적용하기 위해 나타낼 수 있으며, 이를 $\pi_{M-ARC}$ 로 정의한다.

$$\pi_{M-ARC} = \frac{\pi - \omega}{1 - \omega} \quad (4)$$

식 (4)에서 도출한 $\pi_{M-ARC}$ 는 식 (1)의 p를 대체할 수 있다. 식 (1), (4)를 바탕으로 수신자에서의 동작 과정을 나타내면 표 1과 같다. 피드백 메시지를 받은 이 후 송신자에서의 동작 과정은 TFMCC의 과정과 동일하다.

**표 1 수신자에서의 동작 과정**

```
# Receiver's operation
  Foreach (LOSS_EVENT)
    Calculate ω, π_M-ARC ;
    Obtain RTT ;
    # Calcuate T_M-ARC ;
    Feedback to sender ;
  end ;
```

또한, M-ARC의 특징인 전체 패킷 손실률을 인식시키기 위하여 다음 그림 1과 같은 네트워크 아키텍처로 구성 되어진다. 특징은 RTP 또는 RTCP와 연결되어서 제어 피드백을 수신하면 송신자가 수신자들의 패킷 손실률과 연결된 링크의 상황을 판단할 수 있다.

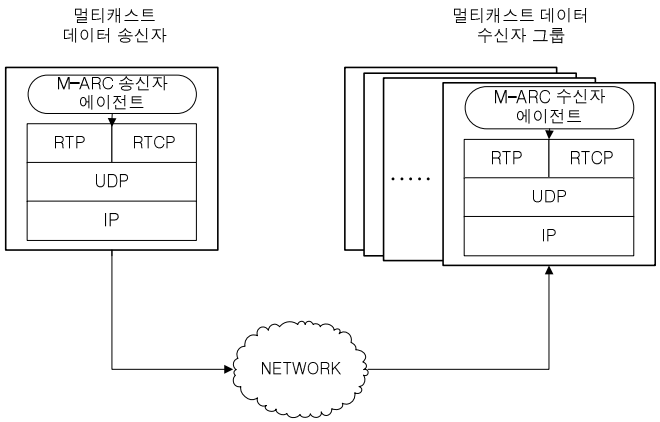


**그림 1 M-ARC의 네트워크 구조**

## 4. 성능 평가

### 4.1. 토폴로지 및 기본 시나리오 구성

M-ARC는 멀티캐스트 도메인에서 무선 환경을 고려하여, 전송률을 제어하는 기법이다. 이를 평가하기 위해, 사용되는 대표적인 bottleneck 링크 모델인 dumbbell 모델을 바탕으로, 그림 2에서와 같이, 데이터가 1개의 송신 노드에서 4개의 수신 노드로 전송되도록 구성하였다.

본 토폴로지에서는 수신 노드를 4개로 구성하였으며, 병목 링크 구간의 용량은 C=1000 packets/s 로 설정하였다. 병목 링크를 제외하고 송신 노드는 유선 링크로 구성하고, 수신 노드로의 마지막 홉은 2개의 노드는 유선, 2개의 노드는 무선 링크로 가정하여 구성하였다.

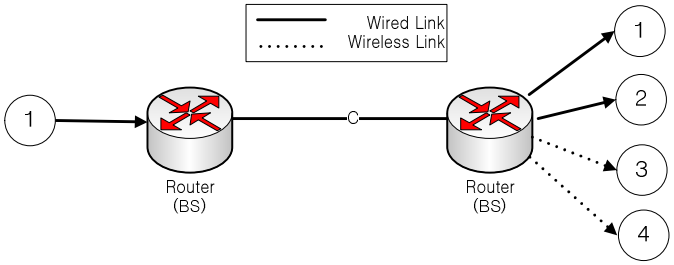


**그림 2 성능 평가를 위한 토폴로지 구성도**

TFMCC와 M-ARC는 멀티캐스트 방식이므로, 각각 좌측 송신 노드가 단일 송신자가 되도록 하였으며, 송신자는 유선 링크를 통해, 라우터(BS)를 거쳐 4개의 수신자에게 데이터가 전달되게 설정하였다. 이때 유선과 무선을 구분하기 하기 위해서 상위의 2개의 유선단과 하위 2개의 무선단으로 구성하였다.

기본 시나리오 구성을 위해 시뮬레이션 시간은 총 120초로 설정하였으며, 백그라운드 트래픽의 설정은 각 1개의 송신 노드에서 4개의 수신 노드로 유니캐스트 프로토콜인 TCP를 기반으로한 FTP 트래픽을 고정적으로 2.0초에서 120초까지 흘려보내도록 하였다.

### 4.2. 성능 평가 수행 및 결과

본 논문에서 성능 평가에 있어, 시뮬레이터 툴은 NS-2 시뮬레이터[7]를 사용하였으며, TFMCC의 NS-2 소스코드는 J. Widmer 의 TFMCC distribution[11]을 사용하여 수행하였다. 시뮬레이션 환경으로, 운영체제는 fedora-9.0 버전 리눅스, 시뮬레이터 버전은 NS-2.30을 이용하였다.

M-ARC의 성능 평가를 위해 비교 대상으로 TFMCC로 설정하고 이를 무선 상황에서 테스트하였다. 성능 평가 방법은 먼저 그림 2를 바탕으로 TFMCC를 적용한 1번 노드를 단일 송신자로 설정하여, 성능을 측정하고, 다음으로 M-ARC를 적용한 1번 노드를 단일 송신자로 설정하여, TFMCC 성능 평가 방법과 동일하게 측정하였다. 성능 평가 비교를 위한 측정 항목으로는 전송률을 기준으로 하였으며, 각각 시간의 흐름과 무선 손실률의 증가를 통해 이를 측정하였다.

그림 3은 120초의 성능 평가 시간 내에서 각각 20초, 50초, 80초에 무선 링크 상의 수신자가 이동으로 인해 무선 환경의 채널 상태가 나빠졌음을 가정하고, 각각 TFMCC와 M-ARC에서 성능 평가를 수행하였을 때, 나타나는 전송률의 변화를 나타내고 있다.

결과를 분석하면, 무선 환경의 채널 상태가 나빠지는

상황인 20초, 50초, 80초에서 TFMCC와 M-ARC 모두 전송률이 감소하였다. 그러나, 전체적인 전송률은 약 20초 이후, M-ARC가 TFMCC에 비해서 약 2배가 높아짐을 알 수 있다.

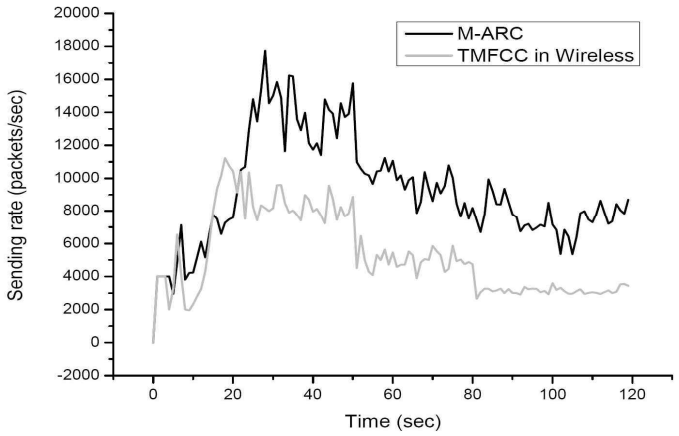


그림 3 시간에 따른 전송률의 변화 비교

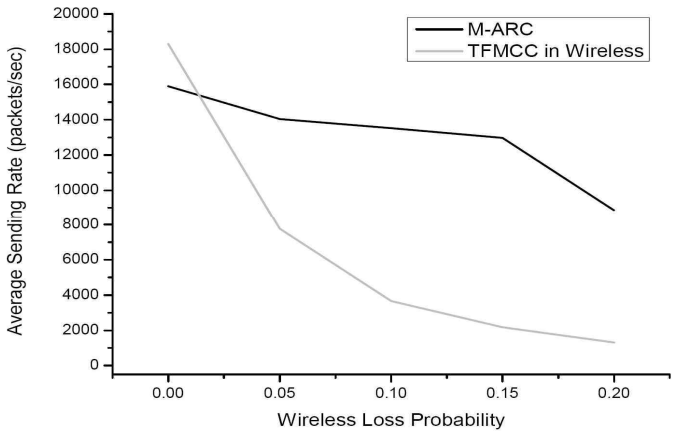


그림 4 무선 손실률 증가에 따른 평균 전송률의 변화

그림 4는 무선 손실률이 증가함에 따라 평균 전송률이 어떻게 변화하는지를 나타낸다. 좀더 일반화시킨 성능 평가를 위해, ω값을 각각 0, 0.05, 0.1, 0.15, 0.2로 변화를 주고, 각 ω값마다 120초 동안의 성능 평가를 5회씩 실행하고 mean 값을 적용시켰다. 그 결과, 무선 손실률이 증가함에 따라 M-ARC가 약 2.6배 더 높은 평균 전송률을 보였다.

무선 환경에서 멀티캐스트 방식을 통해 통신을 하게 되면, TFMCC의 경우, 실제로 네트워크 혼잡 상황이 아님에도 불구하고, 무선 환경을 구분하지 못하고, 가장 낮은 성능을 보이는 수신자의 전송률에 맞추어 전체 전송률을 떨어뜨리게 된다. 그렇기 때문에, 무선 손실률이 증가함에 따라 평균 전송률이 점차 감소하게 되는데, 첫 번째 성능 측정에서는 TFMCC의 성능이 전체적으로 M-ARC의 약 절반의 전송률을 보였으며, 두 번째 성능 측정에서는 기하 급수적으로 감소하는 추세를 볼 수 있었다. 이에 반해, 본 논문에서 제안하는 M-ARC 방식을 동일 조건에서 측정했을 시, 무선 채널 상황이 나빠지거나 무선 손실률이 높아질수록, 전체적인 전송률은 함께 떨어지지만, TFMCC에 비해서, 감소 폭이 매우 적다는 것을 알 수 있다. 따라서, 무선 환경에서 멀티캐스트 방식의 서비스를 제공 받을 시, TFMCC보다 M-ARC를 적용했을 때 좀 더 좋은 전송률을 보장받을 수 있다는 장점을 얻을 수 있었다.

## 5. 결 론

본 논문에서는 유선 환경만을 고려한 멀티캐스트 기반인 TFMCC에서 유선환경과 무선환경을 모두 고려하는 유니캐스트 기반인 ARC의 TCP 전송률 equation에 기초하여, 유무선 환경에서 효율적인 멀티캐스트를 수행하기 위한 M-ARC를 제시하고 성능 평가를 하였다. 기존의 TFMCC는 최저의 성능을 보이는 수신단에 전송률이 맞추어지는 특성을 보인다. 이는 무선 수신단이 멀티캐스트 채널에 가입을 하게 되면, 전체 네트워크의 급격한 성능 저하를 야기한다. 그러나, M-ARC에서는 이런 현상을 회피하기 위해서 채널 가입자의 전체 네트워크 환경을 함께 고려하는 기존 ARC의 TCP 전송률 equation을 적용하였다. 성능 평가는 무선 환경에서 기존의 TFMCC와 M-ARC를 비교하였으며, 그 결과 M-ARC가 시간에 따른 전송률, 무선 손실률에 따른 평균 전송률이 더 높게 나왔음을 볼 수 있었다. 이러한 성능 평가 결과로써 무선 환경에서 멀티캐스트 데이터 전송 시 M-ARC가 TFMCC에 비해 효율적임을 보여주고 있다.

## 참고문헌

[1] S. Floyd, M. Handley, J. Padhye, and J. Widmer. “Equation-based congestion control for unicast applications”, In Proc. ACM SIGCOMM, pages 43 – 56, Stockholm, Sweden, Aug. 2000.

[2] J. Widmer and M. Handley, “Extending equation based congestion control to multicast applications”, in Proc. of ACM SIGCOMM '01,2001.

[3] L. Rizzo. “pgmcc: A TCP-friendly single-rate multicast congestion control scheme”. In Proc. ACM SIGCOMM, pp.17 - 28, Stockholm, Sweden, August 2000.

[4] IETF(The Internet Engineering Task Force), http://www.ietf.org/

[5] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose. “Modeling TCP Reno performance: a simple model and its empirical validation.”, IEEE/ACM Transactions on Networking, 8(2):133–145, April 2000.

[6] O. B. Akan and I. F. Akyildiz, “ARC: the analytical rate control scheme for real-time traffic in wireless networks”, IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 634-644, 2004.

[7] NS-2 simulator, http://www.isi.edu/nsnam

[8] H. Balakrishnan, H. S. Rahul, and S. Seshan, “An integrated congestion management architecture for Internet hosts,” in Proc. ACM SIGCOMM, Sept. 1999, pp. 175–187

[9] S. Cen, C. Pu, and J.Walpole, “Flow and congestion control for Internet media streaming applications,” in Proc. SPIE Multimedia Computing and Networking, Jan. 1998, pp. 250–264.
[10] R. Rejaie, M. Handley, and D. Estrin, “RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet,” in Proc. IEEE INFOCOM, vol. 3, Mar. 1999, pp. 1337–1345
[11] TFMCC distribution,
http://icapeople.epfl.ch/widmer/tfmcc/code/installation.html