

PGMCC 의 Acker 선정 방법 개선

현도원*, 장주욱*

*서강대학교 전자공학과

e-mail : snatcher@ecca1.sogang.ac.kr, jjang@sogang.ac.kr

Improve Acker Selection of PGMCC

Do-won Hyun*, Ju-wook Jang*

*Dept. of Electronic Engineering, Sogang University

요 약

본 논문은 멀티캐스트 혼잡제어방식 제안 중의 하나인 PGMCC 의 개선된 Acker 선정 방법에 관한 것이다. 안정적인 멀티캐스트 전송과 혼잡제어를 위한 여러 가지 방법 중에서 PGMCC 는 유력한 방식으로 제안되었다. 하지만 PGMCC 에도 불확실한 Acker 선정으로 인하여 올바른 Acker 를 선정할 때까지 빈번한 Acker 의 변경이 이루어 지고 이로 인하여 쓰루풋이 낮아지는 문제가 있다. 본 논문에서는 라우터에서 먼저 도착하는 NAK 만을 송신자에게 전달하고 그로 인하여 다른 NAK 를 억압함으로써 발생하는 불확실한 Acker 선정을 개선하기 위하여 송신자에게 도착하는 NAK 를 대상으로 쓰루풋 비교를 통한 개선된 Acker 선정 방법을 도입하여 이와 같은 문제를 해결하고 쓰루풋이 향상 되는 것을 보인다.

1. 서론

멀티캐스트는 다수의 사용자에게 동일한 데이터를 전송할 때 하나의 링크에 동일한 데이터 스트림이 두 번이상 지나가지 않게 함으로서 대역폭을 절감하는 전송 방법으로서 다수에게 멀티미디어 데이터를 전송하는 VOD(Video-On-Demand), 실시간 인터넷 원격강의 시스템과 같은 경우에 데이터의량을 획기적으로 줄일 수 있게 한다[1]. 그러나 기존에 널리 사용되고 있는 TCP(Transmission Control Protocol) 트래픽의 경우에는 혼잡제어가 되는 반면 멀티캐스트 트래픽의 경우에는 혼잡제어가 이루어지지 않기 때문에 같은 인터넷 망에서 공존할 수가 없다. 이러한 문제점을 해결하기 위하여 RMTP(Reliable Multicast Transport Protocol), TFMCC(TCP-Friendly Multicast Congestion Control), PGM(Pragmatic General Multicast) 등의 여러 가지 멀티캐스트 방식들이 제안되었다.

이러한 방식 중에서 PGMCC(Pragmatic General Multicast Congestion Control)[2]는 단일 속도 제어의 대표자 기반 멀티캐스트 혼잡제어 방식이다. PGMCC 는 Acker 라 불리우는 대표자와의 데이터 송,수신을 통해 전송 윈도우의 크기를 조절한다. Acker 는 수신자중에서 가장 낮은 전송율을 가지는 수신자가 Acker 로 선출된다. PGMCC 는 현재 NAK 중심의 멀티캐스트 혼

잡제어 방식 중에서 유망한 제안중의 하나지만 단점을 가지고 있다.

PGMCC 의 경우 라우터에 먼저 도착하는 NAK 만을 송신자에게 전달하고 그로 인하여 다른 NAK 를 억압함으로써 불확실한 Acker 가 선정되는 일이 빈번하게 발생한다. 이러한 불확실한 Acker 의 선정은 올바른 Acker 를 선정할 때까지 빈번한 Acker 의 변경을 가져오게 되며 이러한 것은 PGMCC 의 전체적인 쓰루풋을 낮추는 주요한 원인이다.

본 논문에서는 이러한 단점을 보완하기 위해 새로운 Acker 선정 방법을 추가, 보완 하였다.

2. 기존 PGMCC 의 문제점

2.1 PGMCC

PGM[3]은 단일 송신자 방식의 멀티캐스트 방식으로 NAK 를 기반으로 한 재전송 요청을 통하여 안정적인 멀티캐스트를 제공한다. PGM 에서는 케환 억압을 통하여 멀티캐스트 망의 확장성을 제공한다.

(그림 1)은 PGM 의 시나리오를 보여주고 있다. 송신자는 ODATA(Original Data)를 각 수신자에게 멀티캐스트로 전송한다. 이 때, 데이터를 받지 못한 수신자는 송신자를 향해서 NAK 를 전송하게 된다. 라우터에서는 먼저 받은 NAK 를 위의 라우터로 전송하고 밀

의 수신단에는 더 이상 NAK 를 전송하지 말라는 NCF(NAK Confirmation) 메시지를 전달하게 된다. 만약에 같은 NAK 를 이 라우터가 받았을 경우에는 NAK 를 전송하지 않고 억압하게 된다.

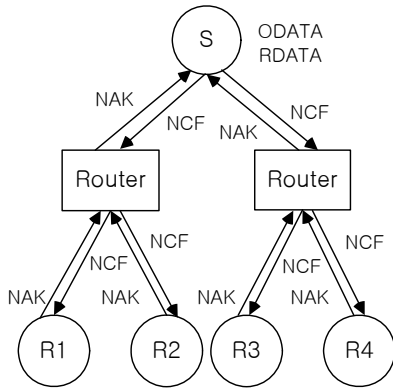


그림 1 PGM 시나리오

PGMCC 는 PGM 에서 혼잡제어를 하기 위하여 제안된 방식이다. Acker 라고 하는 대표자를 선택하여 송신자와 이 대표자 사이에는 제어 관계를 유지한다. 이러한 대표자를 Acker 라고 부르는 이유는 대표자만이 ACK 를 보내기 때문이다. 다른 수신자의 경우 패킷이 손실되었을 경우에만 NAK 를 보낼 수 있다.

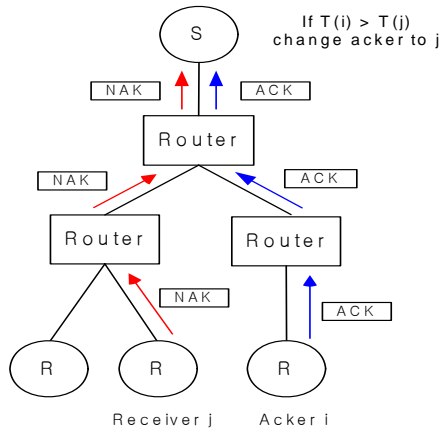


그림 2 PGMCC 시나리오

간단한 시나리오를 (그림 2)에서 확인 할 수 있다. Acker 는 그룹의 대표자로서 가장 나쁜 쓰루풋 (throughput)을 가지는 수신자가 선택된다. 송신자와 Acker 사이의 전송률은 TCP 와 비슷한 윈도우 (window) 기반으로 결정되며, 이는 역시 ACK 에 의하여 결정된다. PGMCC 에서 수신자가 쓰루풋을 판단하는 것은 수신자의 보고에 의해서 결정된다. 수신자는 NAK 나 ACK 에 들어있는 손실률(loss rate)과 시퀀스 번호(sequence number)를 이용하여 RTT 를 측정, 쓰루풋을 결정하게 된다. 손실률은 진동을 피하기 위하여 저대역필터를 사용하여 계산을 한다.

PGMCC 에서 가장 중요한 점은 Acker 의 선택에 있다. Acker 는 가장 낮은 쓰루풋을 가지는 수신자가 선택된다. Acker 의 선택은 쓰루풋의 비교를 통해서 결정한다.

$$T \propto C \frac{1}{RTT \sqrt{p}}$$

T : Throughput

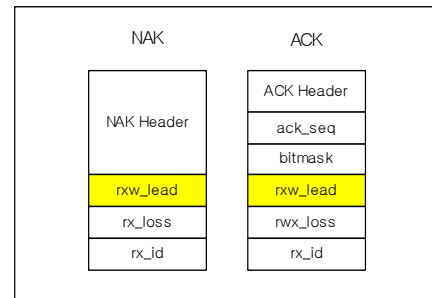
RTT : Round Trip Time

p : Loss rate

C : Constant, PGMCC 의 경우 0.9[2]

이를 계산하여 기존의 대표자 i 와 수신자 j 로부터 받은 NAK 를 기반으로 하여 T(i) > T(j)일 경우 수신자 i 에서 수신자 j 로 대표자가 바뀌게 된다. (그림 3)은 RTT 측정에 대한 그림이다.

Packet from receiver



rxw_lead = receiver's most recently received Sequence number

receiver's rtt = sender's most recently received sequence number - rxw_lead

그림 3 각 수신자의 RTT 측정

RTT 측정을 편하게 하기 위해서 PGMCC 에서는 시퀀스 번호를 사용해서 RTT 를 측정하게 된다. PGMCC 에서의 RTT 는 단순히 쓰루풋 비교를 위해서만 필요하기 때문에 각 수신자의 RTT 를 타임스탬프 (timestamp)를 사용하여 정확하게 측정하는 것이 아니라 시퀀스 번호의 차이로서 결정한다. 이를 위해서 PGMCC 에서는 ACK 나 NAK 에 수신자가 가장 최근에 받은 패킷의 시퀀스 번호를 표기하고 수신자에게서 온 패킷을 받았을 때, 가장 최근에 보낸 패킷과의 차를 RTT 로 사용한다.

2.2 문제점 및 알고리즘을 위한 근거

PGMCC 는 가장 낮은 쓰루풋을 가지는 수신자를 대표자로 선택하여 전송함으로써 TCP 와 친화적인 공정성을 보장할 수 있다. PGM 에서 제안된 제한 억압 방식은 가장 먼저 라우터에 도착한 NAK 를 송신자에게 올려 보내고 다음에 도착하는 NAK 의 경우 송신자에게 전송하지 않는다. 이 때, 패킷 손실률이 비슷한 수신자가 있을 경우, 송신자에 가까운 수신자는 NAK 를 쉽게 송신자에게 보낼 수 있는 반면, 멀리 있는 수신자는 NAK 가 가까이 있는 수신자보다 늦게 도착하여, 송신자에게 NAK 를 전송할 수가 없다. 이 때 멀리 있는 수신자의 쓰루풋은 알 수가 없으므로, Acker 선정에서 제외되게 된다. 이렇게 잘못된 Acker 가 선정되게 되면, 잠시 후에 다시 멀리 있는 수신자에게서 NAK 가 발생하여 송신자까지 올라와서 Acker 가 바뀌게 된다. 이렇게 한번 Acker 를 잘못 선정하게 되면 올바른 Acker 가 선정될 때까지 몇차례의 Acker

선정을 통해서 비로서야 올바른 Acker 가 선정되게 된다. 이러한 빈번한 Acker 의 변경은 PGMCC 의 전체적인 쓰루풋을 낮추는 요인이 된다.

이러한 내용에 대한 근거로서 (그림 4)와 같은 토폴로지에서 실험한 결과를 (그림 5)에서 볼 수 있다. 세로축으로 선이 그어진 부분이 Acker 가 바뀌는 시점이고 216 초 부근의 이러한 부분을 확대해 본 (그림 6)을 보면 Acker 가 변경된 후 곧 바로 다시 Acker 가 변경되는 것을 볼 수 있다. 이러한 현상 앞에서 말한 것 처럼 처음에 올바른 Acker 를 선정하지 못하였기 때문에 발생한다. 그리고 Acker 가 변경되면서 쓰루풋 또한 낮아지는 것을 볼 수 있다.

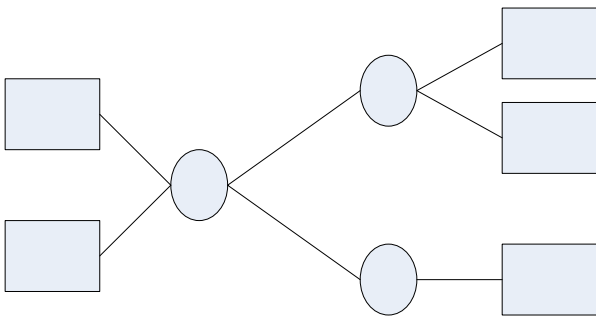
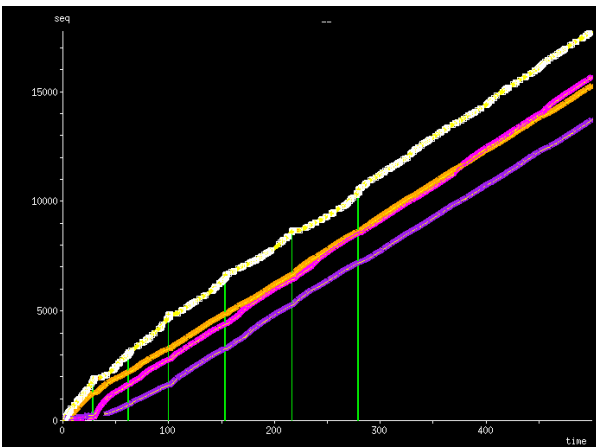


그림 4 문제 확인을 위한 실험 토폴로지



PGMCC (Sender) 실험 결과

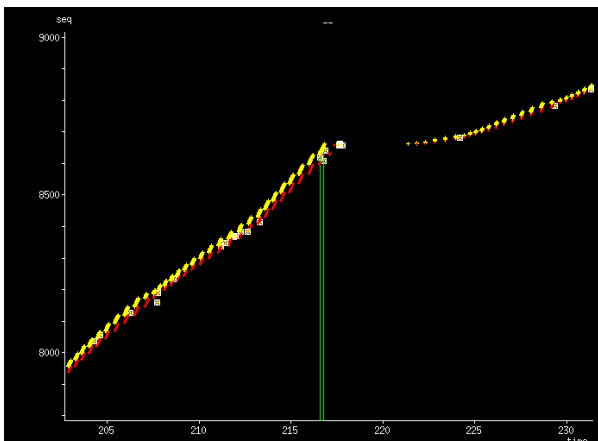


그림 6(그림 5)의 216 초 부근 확대 결과

(그림 5)에서 보면 자세하게는 안보이지만 Acker 가 변경되는 모든 시점에서는 거의 항상 2~3 차례씩 Acker 의 변경이 이루어 진다. 이러한 Acker 의 변경 시간은 토폴로지와 수신자의 수에 따라서 다르지만 평균적으로 0.76 초 정도 되는 것을 실험적으로 확인 하였다.

3. 제안된 방법

기존의 PGMCC 에서는 송신자가 NAK 를 받으면 현재의 Acker 와 NAK 를 보낸 수신자간의 쓰루풋을 비교하여 현재의 Acker 를 유지할 것인가 또는 새로 NAK 를 받은 수신자를 Acker 로 받을 것인가를 결정하여 Acker 를 선정한다. 하지만 본 논문에서 제안하고자 하는 방법은 송신자가 NAK 를 받고서 바로 NAK 를 보낸 수신자와 Acker 사이의 쓰루풋을 비교하여 새로운 Acker 를 선정하는 것이 아니라, 수신자가 NAK 를 받게 되면 일정 시간 T 를 기다리고, 그 시간 안에 송신자가 새로운 NAK 를 받으면 다시 일정 시간 T 를 기다린다. 이렇게 송신자가 새로운 NAK 를 기다리다가 일정시간 T 안에 더 이상 새로운 NAK 가 오지 않으면 지금까지 온 NAK 들과 현재 Acker 사이의 쓰루풋을 비교하여 송신자가 받은 NAK 중에서 쓰루풋이 가장 낮은 수신자와 Acker 사이의 쓰루풋을 비교하여 쓰루풋이 낮은 수신자를 새로운 Acker 로 선정한다.

(그림 7)는 개선된 새로운 Acker 선정 방법의 알고리즘 순서도이다.

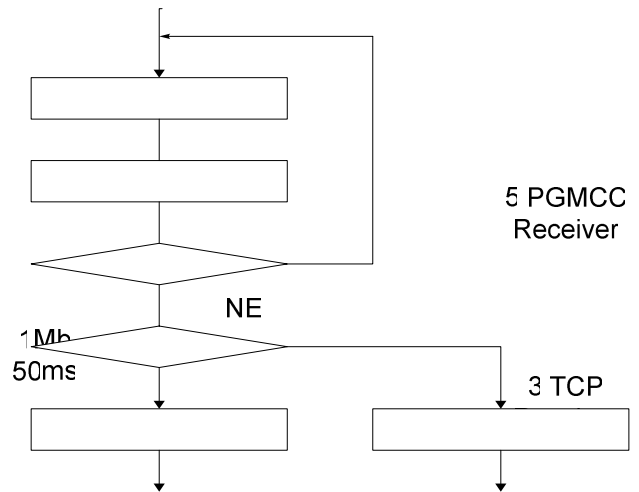


그림 7 개선된 Acker 선정 알고리즘

5Mb 420 실험결과 및 결과분석

본 논문에서는 제안한 알고리즘의 검증을 위하여 NS-2 와 C++을 사용하여 실험하였다.

(그림 8)과 같은 시뮬레이션 토폴로지에서 기존의 같 PGMCC 와 개선된 Acker 선정 방법이 적용된 PGMCC 의 쓰루풋을 측정해 보았다.

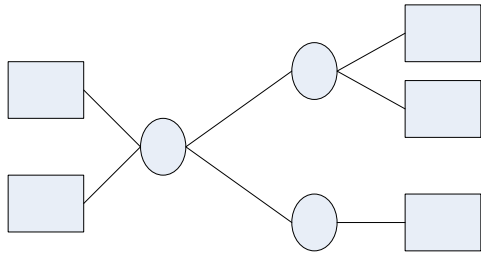


그림 8 시뮬레이션 토폴로지

개선된 PGMCC 에서 송신자가 NAK 를 기다리는 시간은 앞에서 말한 평균값인 PGMCC 초를 사용하였다. (그림 9)는 기존의 PGMCC 와 제한한 Acker 선정 알고리즘을 적용한 각각의 쓰루풋을 나타낸 것이다.

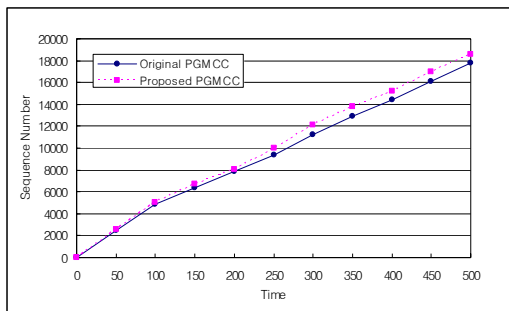


그림 9 개선된 Acker 선정 방법이 적용된 PGMCC 와 기존의 PGMCC 의 쓰루풋 비교

위의 (그림 9)에서 보듯이 개선된 Acker 선정 방법을 적용 하였을 경우 기존의 PGMCC 보다 약 5% 정도 쓰루풋이 향상되었음을 확인 할 수 있다.

이번에는 멀티캐스트 트리의 깊이를 늘려서 (그림 10)과 같은 토폴로지에서도 실험을 하였다.

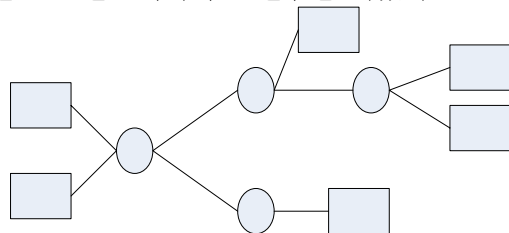


그림 10 시뮬레이션 토폴로지

앞의 실험과 마찬가지로 송신자가 NAK 를 기다리는 시간은 0.76 초로 설정하여 실험하였다.

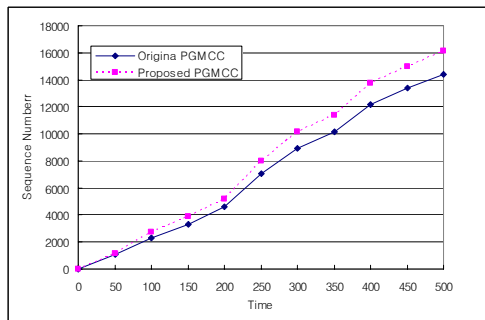


그림 11 개선된 Acker 선정 방법이 적용된 PGMCC 와 기존의 PGMCC 의 쓰루풋 비교

(그림 11)에서 보면 앞의 실험보다 더 나은 약 10%의 쓰루풋 향상을 볼 수 있다. 이러한 원인은 (그림 8)의 토폴로지보다 (그림 10)의 토폴로지에서도 실험을 하였을 경우 더 많은 Acker 의 변화가 있어서 (그림 11)과 같이 쓰루풋이 더 향상 되었다.

5. 결론

PGMCC 는 NAK 억압 방식을 사용하여 PGMCC 에 의한 폭주 현상을 방지하고 수신자 중에서 수신률이 가장 낮은 수신자인 Acker 를 선정하여 전송을 한다. 하지만 라우터에서의 불확실한 NAK 억압으로 인해서 올바른 Acker 를 선정하는데 있어서 문제가 있고 이로 인하여 쓰루풋이 낮아지는 문제가 있었다. 본 논문에서는 송신자에서 올바른 Acker 를 바로 선정하고 그로 인하여 쓰루풋의 향상을 보여 주었다.

추후 과제로는 시뮬레이션이 아닌 실제 네트워크망에서의 실험과 올바른 Acker 를 선정하고 PGMCC 의 성능을 높이기 위하여 본 논문에서 지적한 개선된 Acker 선정 방법을 적용시키는 것이 있다.

참고문헌

- [1] Chwan-Hwa Wu, J. David Irwin, "Emerging Multimedia Computer Communication Technologies," Prentice Hall, 1988.
- [2] L. Rizzo, L. Vicisano, M. Handley and G. Iannaccone, "PGMCC single rate multicast congestion control: protocol specification," Internet Draft IETF, Feb., 2001
- [3] Tony Speakman, Dino Farinacci, Jon Crowcroft, Dan Leshchiner, Luigi rizzo, "PGM Reliable Transport Protocol Specification, Internet-Draft, draft-speakman-pgm-spec-05.txt," November, 2000.
- [4] Chin-ying Wang and Sonia Fahmy, "Dynamics of the "pgmcc" Multicast Congestion Control protocol." Tektronix Report, August, 2001.
- [5] W. Richard Stevens, "TCP/IP Illustrated, Volume I," Addison-Wesley, 1994.
- [6] J. Padhye, V. Firoiu and J. Kurose, "Modeling TCP throughput: A simple Model and its Empirical Validation," ACM SIGCOMM 1998, Vancouver, BC, Canada, September, 1998.
- [7] Luigi Rizzo, "pgmcc: a TCP-friendly single-rate multicast congestion control scheme," ACM SIGCOMM 00, 2000.
- [8] S. Floyd, M. Handley, J. Padhye and J. Widmer, "Equation-based congestion control for unicast Applications," Proc. ACM SIGCOMM. pp.43-56, Aug., 2000.
- [9] Sneha K. Kasera, Supratik Bhattacharyya, Mark Keaton, "Scalable Fair Reliable Multicast Using Active Services." IEEE/ACM Transactions on Networking, Aug., 2000.
- [10] 박영선, 현도원, 장주욱, "PGMCC 의 공정성 향상," 정보처리학회논문지 C 제 10-C 권 제 3 호, Jun. 2003.