ECN 마킹을 위한 적당한 Threshold

임재걸[O] 이계영 장익현 윤준영
동국대학교 경주캠퍼스 컴퓨터학과
{yim[O], lky, ihjang, yoon5237}@dongguk.ac.kr

Optimal Threshold for ECN Marking

Jaegeol Yim[O] Gyeyoung Lee, Ikhyeon Jang, Joonyoung Yoon
Dongguk Univ. Computer Science

요 약

무선 네트워크의 TCP 성능을 제고하기 위한 방법으로 ECN 방법이 심도 있게 연구되었다. 그러나, ECN 마킹을 위한 최적의 threshold에 대한 연구 결과는 찾아 보기 힘들다. 본 논문은 ECN 마킹을 위한 최적의 threshold를 찾는 식을 제시하고, ECN 전략을 적용하는 TCP의 페트리 넷 모형을 구축한 다음, 시뮬레이션을 통하여 제시된 식의 타당성을 검증한다. 또한, 제시된 식의 실용화 방안도 제안한다.

1. 서 론

트랜스미션 콘트롤 프로토콜 (TCP: Transmission Control Protocol)은 프로세스간의 신뢰성 높은 연결을 구축해 주는 통신 규약으로, 현재 인터넷 통신량의 대부분이 TCP 정보이다. TCP 규약은 연결부와 통신부로 구성되어 있다. 연결부는 연결을 설정하기 위하여 거쳐야 할 과정과 연결을 종료하기 위하여 거쳐야 할 과정으로 구성되며 비교적 간단하기 때문에 TCP 의 성능 연구는 보통 통신부에 초점을 둔다.

TCP 통신은 송신 TCP 가 수신 TCP 에게 일정량의 세그먼트를 전송하고, 수신 TCP 가 접수확인을 보내오면 송신 TCP 가 다음 세그먼트들을 일정량 전송하는 방식으로 진행된다. 이때 접수확인을 받기 이전에 한꺼번에 보내는 데이터량을 윈도우라고 칭한다.

TCP 통신부의 가장 중요한 컴포넌트는 통신 혼잡 제어부분으로, 송신자가 수신자의 접수확인을 수신하면 윈도우의 크기를 증가시키는 반면에, 접수확인이 일정시간 이내에 도착하지 않으면 Timeout 윈도우의 크기를 감소하는 방법으로 통신혼잡을 제어한다. Timeout 이 발생하는 원인은 해당 세그먼트가 통신오류로 손실되든지 아니면 어느 라우터에서 버퍼 넘침으로 drop 된 경우이다. 유선 네트워크에서는 Timeout 의 원인을 후자로 해석하고 윈도우의 크기를 감소시키는 것이 적당한 혼잡제어 방식이지만, 무선 네트워크에서는 통신오류로 기인한 timeout 을 통신 혼잡으로 오해하고 윈도우의 크기를 감소함으로써 통신 용량을 충분히 사용하지 않는 결과를 낳는다.

무선 네트워크에서 이와 같이 통신 용량을 허비하는 것을 방지하는 방법으로 '혼잡 알림' (ECN: Explicit Congestion Notification) 방식이 있다. 이 방식은 라우터나 게이트웨이에서 버퍼가 어느 정도 차서 drop 이 예상되면 세그먼트에 ECN 마킹을 하여줌으로써 송신 TCP 가 통신혼잡을 감지하고 윈도우의 크기를 조절하게 하는 것이다. 이 방식을 적용하기 위하여 풀어야 할 문제는 '과연 버퍼가 어느 정도 찼을 때 ECN 마킹을 하는 것이 바람직한가'이다. 본 논문은 이 문제에 대한 답을 제시하고, 답의 타당성을 보이기 위하여 시뮬레이션 결과를 제공한다. 시뮬레이션을 위하여 ECN 방식을 채용한 TCP 의 페트리 넷 모형을 구축한다.

2. 관련연구

전통적으로 통신 혼잡은 세그먼트 손실로 감지되었다. 무선통신에서는 통신 혼잡에 의한 세그먼트 손실보다 무선통신 회선에서 발생하는 오류에 의한 세그먼트 손실이 더 빈번함에도 불구하고 TCP 는 이들을 통신 혼잡으로 간주함으로써 통신 효율성을 크게 감소시키는 단점이 있다. 이러한 단점을 보강하는 무선통신 환경의 통신 혼잡 회피 방법은 크게 '혼잡 알림' (ECN: Explicit Congestion Notification), '통신오류 알림' (Explicit Wireless Loss notification) 그리고 '단대단 방식'으로 대별된다.

ECN 방법은 통신 혼잡 상황을 TCP 통신 경로상의 라우터나 게이트웨이가 세그먼트나 패킷에 표시하는 것이다. 이러한 방법으로는 '원천 해소 메시지(SQM: Source Quench Message)'[1-5], DECbit [6,7], RED(Random Early Detection) [8] 등이 있으며, 이들 세 가지 방법 외에도 나름대로 독특한 방법을 고안하여 그냥 ECN 방법이라는 이름으로 제안한 사례가 무수히 많다.

ECN 방법의 기본 원칙은 라우터나 게이트웨이에서 통신혼잡이 발생하면 이 사실을 송신 TCP 에게 알려줌으로써 트래픽을 적당히 조절할 수 있도록 하는 것이다. 문제는 '과연 무엇을 기준으로 통신혼잡의 발생을 판단할 것인가'이다. 이에 대한 답으로 [9]는 최초의 drop 을 사용하였다.

이에 대하여, 본 논문은 다음에 보이는 식 1 로 구하는 Threshold 를 사용할 것을 제안한다. 즉, 라우터의 버퍼에 들어 있는 세그먼트의 수가 식 1 에 제시된 threshold 를 넘을 때 ECN 마크를 부착하는 것이 가장 효율적이라는 것이다.

버퍼길이 - ((RTT/TBA) - (RTT/TPS)) = Threshold --- 식 (1)
단, RTT = round trip time : 세그먼트가 송신자를 출발하여 수신자에게 접수되고, 접수확인 세그먼트가 다시 송신자에게 수신되는 데 걸리는 시간.
TBA : 라우터에 i+1 번 세그먼트가 도착한 시각
- i 번 세그먼트가 도착한 시각의 평균
TPS : 라우터가 하나의 세그먼트를 처리하는 데 필요한 시간의 평균.

사용하는 Threshold값이 식 1보다 작으면 통신회선 용량을 다 사용하지 않고 혼잡 윈도우 값을 감소시키는 결과를 낳고, 반대로 Threshold 값이 식 1보다 크면 drop tail이 발생하여 재전송을 하게 되기 때문에 식 1이 가장 적합한 Threshold인 것이다.

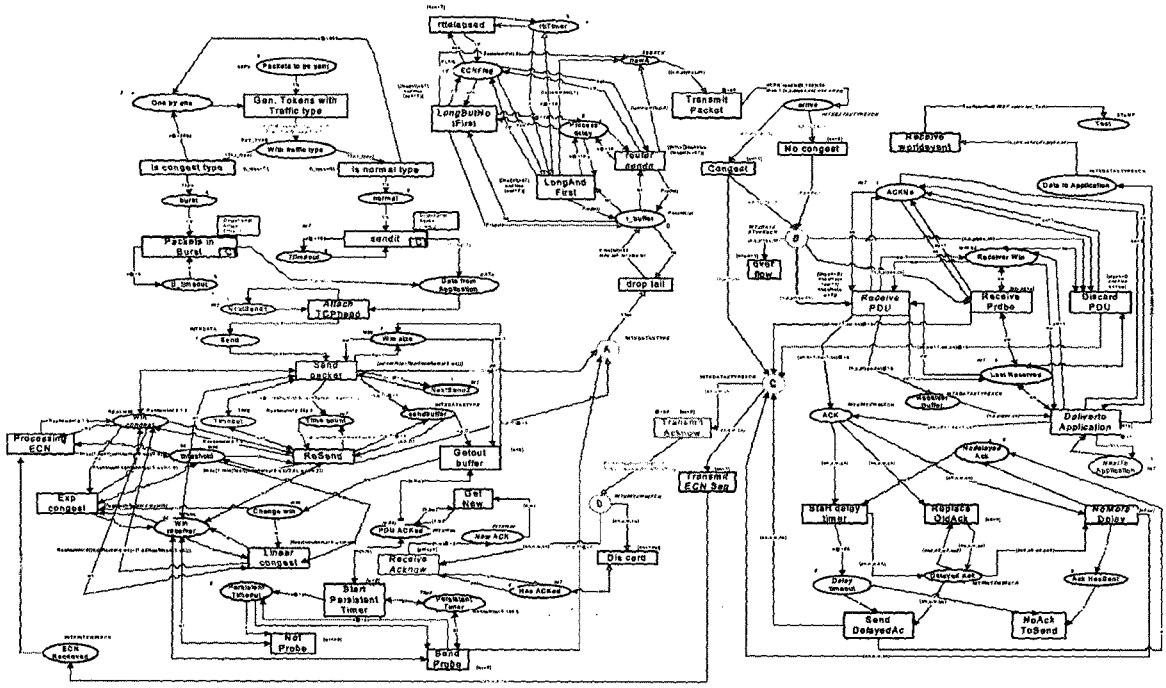

그림 1 A Pettri net model for TCP

3. TCP 의 페트리 넷 모형

페트리 넷과 TCP에 대한 설명은 지면 부족으로 생략하고, 본 실험을 위하여 작성한 페트리 넷 모형을 그림 1에 소개한다. 그림 1은 본 논문의 중요한 공헌 중 하나이지만, 지면 부족으로 설명을 생략한다.

4. 실험

4.1 전송 오류율 0%

본 실험의 목적은 라우터가 ECN 마킹을 하는 기준으로 사용될 Threshold를 구하는 식1의 타당성을 보이는 것이다. Drop tail이 반드시 발생하고 분석하기 쉬운 경우부터 고려하기 위하여 실험 환경을 다음과 같이 설정하였다:

1) 송신 TCP 는 매 5 micro-second 마다 한 세그먼트를 전송하여 총 300 segments 를 전송하고 종료함.
2) transmission error rate 를 0 으로 함.
3) router 버퍼의 크기를 80 으로 함.
4) drop tail 을 틀림 없이 발생시키기 위하여, Congestion Avoidance 를 위한 threshold 를 64 로 함.
5) 전송 시간은 양방향 모두 50 으로 함.

실험 결과 표 1 과 같은 throughput 과 propagation delay 를 얻었다. Noecn, Th50, Th67 은 각각 ecn 을 채용하지 않은 경우, threshold 를 50 으로 한 경우, 그리고 threshold 를 67 로 한 경우를 의미한다. Threshold 67 은 식 1 에 의하여 얻은 값이다.

양방향 전송 시간과 라우터의 처리시간, 송신 TCP 와 수신 TCP 의 처리시간의 합이 120 micro-second 임으로 300 segments 를 하나씩 전송하고, 접수확인을 받으면 또 하나씩 전송하여 300 개를 모두 전송하는데 걸리는 시간은 300*120 = 36,000 임으로, ECN 방식을 채용하지 않은 TCP 는 36,000/10,163 = 3.54 speed up 을 보인다. 예상한 바와 같이 threshold 를 67 로 한 경우에 throughput 과 propagation delay 모두 더 좋다는 것을 알 수 있다. 각 세그먼트에 대한 Propagation delay 는 그림 2 에 보이는 차트와 같다.

실험 결과, Noecn, Th50, Th67 의 성능 차이가 미미한 것을 볼 수 있다. 그 이유는 그림 3에 보이는 바와 같이, Noecn이나 Th50이나 모두 비슷한 횟수의 slow start를 하기 때문이다. 차이점은 Noecn의 경우에는 drop tail 발생 후 Timeout 기간 동안 전송했던 세그먼트들을 재전송하는데 반하여, Th50과 Th67은 재전송 없이 slow start를 시작한다는 것이다. Th50의 경우에는 너무 일찍 ecn 마킹을 함으로써 통신회선의 용량을 충분히 사용하지 않기 때문에 Th67과 미미하지만 성능 차이가 나타나는 것을 알 수 있다.

표 1 전송 오류율이 0일 경우, ECN 마킹을 위한 Threshold 값에 따른 성능 비교

성능지표 \ Threshold	noecn	Th50	Th67
300 개 전송에 걸린 시간	10,163	9,768	8,052
propagation delay	3583.24	3290.2	2754.9

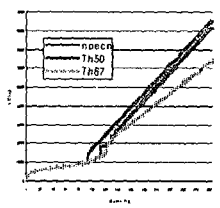

그림 2

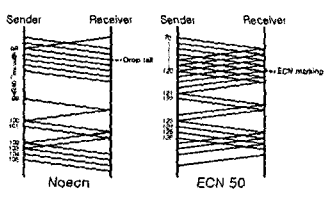

그림 3

그림 2 전송오류율 0 인 경우의 Propagation Delay 비교
그림 3 Noecn에서는 재전송이 발생하는데 반하여, ECN에서는 재전송이 발생하지 않음.

4.2 전송오류율 20%

실 세계에서는 전송 오류가 빈번하게 발생한다. 특히, 무선 네트워크의 전송 오류율은 매우 높다. 실세계 환경을 반영하기 위하여 전송 오류율을 20%로 하여 300 개의 세그먼트를 모두 전송하는 데 걸리는 시간을 측정하여 표 2 와 같은 결과를 얻었다. 앞에서와 마찬가지로, noecn, Th50, Th67 은 각각 ecn 마킹을 사용하지 않는 경우, 라우터 버퍼에 세그먼트가 50(67)개일 때 ecn 마킹을 하는 경우를 나타낸다.

실험 결과 Th50 과 noecn 을 비교할 때, throughput 은 22% 개선 효과가 있으며, 평균 propagation delay 는 42% 개선 효과를 보인다. Th50 과 Th67 을 비교할 때 Th50 의 경우가 성능 우세를 보인다. 이것은 전송오류에 따른 재전송으로 말미암아, 라우터에 도착하는 세그먼트의 빈도가 높아짐에 따라 식 1 의 값이 67 보다 50 에 더욱 가깝기 때문이다.

표 2 전송 오류율이 0.2일 경우, ECN 마킹을 위한 Threshold 값에 따른 성능 비교

성능지표 \ Threshold	noecn	Th50	Th67
300 개 전송에 걸린 시간	25,900	20,270	21,148
propagation delay	11,921.26	6,865.78	7,459.69

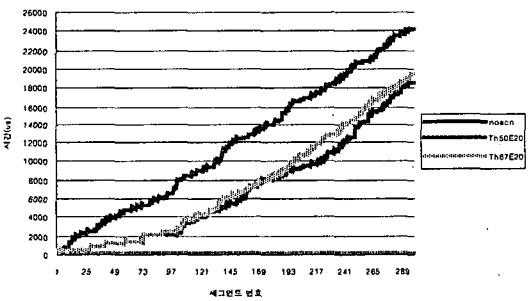

그림 4 전송오류율 0.2 인 경우의 Propagation Delay 비교

5. 적용 방법

앞 절에서, ECN 마킹을 위한 적당한 threshold 를 구하는 식으로 식 1 을 제시하고, 실험을 통하여 이의 타당성을 보였다. 식 1 을 실제에 적용하려면 TCP connection 에 할당된 라우터 버퍼의 길이, RTT, TBA, TPS 를 라우터가 알아야 한다. 라우터의 버퍼 운용 전략은 매우 복잡한 문제이지만, 운용 전략이 결정되면 주어진 TCP connection 에 할당된 라우터 버퍼의 길이를 측정하는 것은 가능하다.

RTT 는 송신자가 세그먼트에 부착하여 라우터에게 알려주어야 한다. Option 필드에 RTT 를 기록하고, 이 사실을 예약 필드에 표시하여 줌으로써 라우터에게 필요한 정보를 전달할 수 있다. TBA 와 TPS 는 라우터에서 통계적으로 비교적 쉽게 구할 수 있는 값이다. 이러한 값들을 동적으로 구하여 식 1 을 평가함으로써 가장 효율적인 ECN 마킹 전략을 운용할 수 있다.

6. 결론

본 논문은 ECN 마킹을 위한 적당한 threshold를 구하는 식으로 식1을 제시하고, ECN 마킹을 적용하는TCP 모형을 페트리 넷으로 구축한 다음, 시뮬레이션을 통하여 식1의 타당성을 보였다. 나아가서, ECN 방법이 전송오류가 높은 네트워크 환경에서 시스템 성능을 제고한다는 사실을 시뮬레이션으로 보이고, 또한, 식1의 실용화 방안도 제시하였다.

여러 가지 실험을 통하여, 본 논문에 소개된 TCP 모형이 실제 TCP를 잘 반영한다는 것을 검증하였다. 또한, 본 모형은 각 사건에 대하여 어느 세그먼트에서 그 사건이 발생하였는지 기록하는 것과 같은 비교적 세밀한 수준의 모형으로 쉽게 변화될 수 있는 장점이 있다. 따라서, 컴퓨터 네트워크를 전공하는 사람들에게 좋은 시뮬레이션 도구로 사용될 수 있으리라고 믿는다.

제안된 방식의 단점으로, 이 방식은 RTT 가 매우 큰 환경에서는 사용될 수 없다는 것을 들 수 있다. 예를 들어 위성통신을 사용하는 환경에서는 RTT 가 0.25 초일 수 있다. TBA 가 100μs 이고 TPS 가 200μs 이라면, 식 1 의 값은 250,000/100 - 250,000/200 = 1,250 이 되어 실용성이 없을 만큼 너무 큰 수가 된다. 근거리 네트워크에서는 제안된 방식이 시스템 성능 개선을 위하여 적용될 수 있다.

참고문헌

[1] Braden, R., and Postel, J., "Requirements for internet gateways," Request for Comments (Standard) RFC 1009, IETF, June 1987, http://www.ietf.org/rfc.html.

[2] Lin, T.Y.Chen, Y.C., "A congestion control approach for LAN/MAN interconnection via ATM," Proceedings of IEEE 13th Networking for Global Communications, Vol. 2, 06/12 - 06/16, 1994, Toronto, Ont.,pp. 892-901.

[3] Dracinschi, A., Fdida, S.,"Congestion avoidance for unicast and multicast traffic," 1st European Conference on Universal Multiservice Networks, 2000, 10/02 - 10/04, 2000, Colmar,, pp. 360-368.

[4] Chen, Z., Jiang, S., Zheng, S., and Ko, C., "Performance comparison between ECN and BECN," Joint 4th IEEE International Conference on ATM (ICATM 2001) and High Speed Intelligent Internet Symposium, 2001, 04/22 - 04/25, 2001, Seoul,South Korea, pp. 305-309.

[5] Peng, F., Leung, V.C.M., "Fast backward congestion notification mechanism for TCP congestion control," 21st IEEE International Performance, Computing, and Communications Conference, 3-5 April 2002, pp. 419 424.

[6] Ramakrishnan, K.K., and Jain, R., "A Binary Feedback Scheme for Congestion Avoidance in Computer Networks," ACM Transactions on Computer Systems, Vol. 8 No. 2, pp. 158-181, 1990.

[7] Jain, R., "Congestion control in computer networks: issues and trends," IEEE Network, Vol. 4, Issue: 3, May 1990 pp. 24 30.

[8] Floyd, S., Jacobson, V., "Random early detection gateways for congestion avoidance," IEEE/ACM Transactions on Networking, Vol. 1, Issue: 4, Aug. 1993, pp. 397 413.

[9] Fei Peng, Shiduan Cheng, Jian Ma, "An Effective way to improve TCP performance in Wireless/mobile Networks," EUROCOMM 2000. Information Systems for Enhanced Public Safety and Security. IEEE/AFCEA, 17 May 2000. pp. 250-255.