공정성 향상을 위한 TCP Vegas의 개선

이정재[0]* 채현석** 최명렬*
한양대학교 전자전기제어계측공학과*, 동원대학교 인터넷정보과**
delphini[0]@asic.hanyang.ac.kr*, hschae@tongwon.ac.kr**, choimy@asic.hanyang.ac.kr*

Modify of TCP Vegas for improvement of fairness

Jung-Jae Lee[0]*, Hyun-Seok Chae**, Myung-Ryul Choi*
Dept. of EECI, Hanyang University*, Dept. of internet & information retrieval, Tongwon College**

요 약

TCP Vegas는 TCP Reno에 비해 효율성과 안전성 측면에서 보다 나은 성능을 보이고 있으나, 두 버전이 같이 사용되는 경우 충분한 성능을 발휘하지 못하여 공정성에 문제점을 드러내고 있다. 본 논문에서는 TCP Vegas의 알고리즘에서 사용하는 상수 요소인 α, β 값의 변화에 의해 TCP Reno 버전과의 공정성이 향상되는 사실을 시뮬레이션을 통해 확인하였으며, 이에 따른 향후 연구과제를 제시하였다.

1. 서 론

TCP는 인터넷 환경에서 가장 널리 사용되고 있는 프로토콜의 하나이다. TCP는 흐름제어를 통해 데이터의 손실 및 재전송을 줄여 효율성을 높일 수 있으며, 이러한 흐름 제어는 데이타 패킷 크기의 제한을 두는 윈도우 기법을 통해 이루어지고 있다. 흐름 제어 방식에 의해 TCP는 Tahoe, Reno, Vegas와 같은 여러 버전으로 구분된다. 이 중, TCP Reno는 고속 복구 기능을 가진 혼잡 방지 이론을 포함하고 있으며 가장 널리 사용되고 있으나, 데이터 패킷의 손실을 감지하여 네트워크의 상황을 판단함으로 인해 네트워크의 혼잡 및 불안전성을 유발하는 문제점을 가지고 있다. TCP Vegas는 혼잡제어 알고리즘을 포함하여 이러한 문제점을 해결하였다. TCP Vegas는 TCP Reno에 비해 37%~71% 높은 효율성을 가지고 있으며, 대역폭 배분의 공정성 및 출력의 안전성에서도 TCP Vegas는 TCP Reno에 비해 보다 나은 성능을 보여주고 있다[1].

그러나 TCP Reno는 패킷의 손실이 발생할 때까지 계속적으로 대역폭을 확장하는 반면, TCP Vegas는 네트워크의 상황에 맞는 적절한 대역폭을 판단하여 대역폭의 확장 및 유지가 이루어지기 때문에 두 버전이 동시에 사용될 경우 TCP Reno가 TCP Vegas에 비해 보다 넓은 대역폭을 차지하여 두 버전간의 공정성 문제가 발생하게 되어 [2][3] 최근 이러한 공정성 문제를 해결하기 위한 연구가 계속되고 있다 [4][5]. 본 논문에서는 실제 인터넷 환경을 가정한 시뮬레이션을 통해 TCP Vegas 알고리즘의 상수 요소인 α, β 값의 변동에 따라 두 버전의 공정성이 향상되는 모습을 확인하고 이에 따른 향후 연구방향을 제시하였다.

2. TCP Vegas

TCP Tahoe와 TCP Reno는 네트워크의 병목현상에 의한 데이터 패킷의 손실을 감지함으로써 네트워크 상에서의 혼잡을 인지한다. 두 버전은 데이터 패킷의 손실을 감지하기 전까지 congestion windows를 계속적으로 증가시키고, 손실이 발생할 경우 windows 크기를 감소시킨 후 손실된 패킷의 재전송이후에는 다시 windows의 크기를 증가시키게 된다.

이와 달리 TCP Vegas는 네트워크의 혼잡을 기본적으로 RTT를 통해 인지하고 있다. 이는 TCP Vegas의 네트워크 혼잡 인식 방식이 extra data의 측정과 그에 따르는 cwnd의 조절에 의해 이루어지기 때문이다. extra data란 라우터의 버퍼에 머무르고 있는 전송중의 데이터 조각을 의미한다. 기본적인 개념은 extra data의 총합이 너무 작거나 너무 크지 않아야 한다는 것이다. 만약 extra data의 합이 β보다 커지게 된다면 데이터의 크기가 버퍼의 용량을 넘어서서 overflow가 발생하게 되고 TCP Vegas는 cwnd를 감소시키게 된다. 이와 달리 extra data의 합이 α보다 작아질 경우 하나의 RTT마다 cwnd는 증가하게 된다. extra data의 합이 α와 β 사이에 있을 경우 cwnd는 같은 값으로 유지하게 된다. TCP Vegas에서는 위와 같이 각각의 RTT를 이용하여 윈도우 크기를 결정한다.

BaseRTT를 queuing delay가 없는 RTT의 최소값이라고 하고, cwnd의 크기가 W로 주어졌을때 네트워크의 혼잡이 없는 상태에서 기대할 수 있는 처리능력은 다음과 같다.

$$Expected = \frac{W}{BaseRTT} \qquad [1]$$

실제 처리능력은 다음과 같이 주어진다.

$$Actual = \frac{W}{RTT} \qquad [2]$$

BaseRTT는 RTT의 최소값이기 때문에 실제 처리능력은 이상적인 처리능력보다 클 수 없다. TCP Vegas는 혼잡회피 결정에 두 값의 차이를 이용하고 있다.

$$diff = Expected - Actual \qquad [3]$$

이때, TCP Vegas는 다음과 같이 판단을 한다.

$$W_{next} = \begin{pmatrix} W_{cur}+1, diff.BaseRTT < \alpha \\ W_{cur}-1, diff.BaseRTT > \beta \\ W_{cur}, \alpha \le diff.BaseRTT \le \beta \end{pmatrix} \qquad [4]$$

여기서 W_{cur}은 기존의 cwnd의 크기이고, W_{next}는 다음 RTT에서의 cwnd의 크기를 말한다. 위의 [4]식에 의해서 α, β 값에 의해 TCP Vegas가 가지는 cwnd값이 결정되게 된다.

3. TCP Vegas와 TCP Reno의 공정성 확보

3.1 시뮬레이션 구성

앞에서 살펴본 바와 같은 이유로 TCP Vegas와 TCP Reno가 같이 사용될 경우 두 버전간의 공정성 문제가 발생하게 된다. 본 논문에서는 두 버전간의 공정성 문제를 해결하기 위하여 TCP Vegas 알고리즘에서 사용하는 상수 α, β 크기를 다양하게 변화시키고 이를 TCP 연구에서 다양하게 사용되고 있는 ns (network simulator)를 사용하여 시뮬레이션을 통해 두 버전간의 공정성 변화를 살펴보았다. 그림 1은 시뮬레이션에 이용한 네트워크 구성도이다.

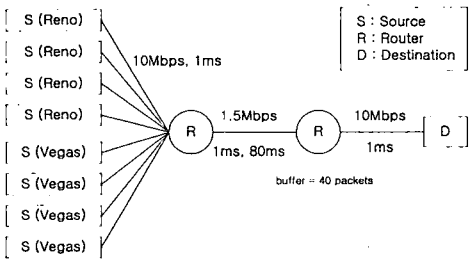


그림 1. 시뮬레이션 네트워크 구성도

각각의 FTP source는 512KB의 고정된 크기를 갖는 데이터 패킷을 10Mbps의 전송용량과 1ms의 전송 delay를 통해 전송하게 된다. 40패킷의 버퍼크기를 갖는 라우터는 source로부터 전송된 데이터 패킷을 1.5Mbps의 전송용량과 1ms와 80ms의 전송 delay를 통해 목적지로 전송하게된다. 이때 네트워크에는 병목현상이 발생하게 된다.

3.2 α, β값의 변화를 통한 공정성 향상

라우터에서 목적지로 데이터가 전송될 때 delay가 1ms로 적은 경우 시뮬레이션 결과는 그림 2와 같다.

delay가 적어 데이터 패킷의 손실 및 혼잡이 발생하지 않는 경우 α=1인 경우 TCP Vegas와 TCP Reno의 공정성이 가장 좋음을 알 수 있다. α>2 이상인 경우 오히려 TCP Reno가 TCP Vegas에 비해 공정성이 떨어지는 모습을 보였다. α와 β의 차이가 4이상인 경우 α와 β 차이에 의한 공정성의 변화는 일어나지 않았다.

실제 인터넷 환경을 고려하여 delay를 80ms로 증가시킨 경우 그림 3에서와 같이 α값이 증가함에 따라 TCP Vegas와 TCP Reno의 공정성이 향상됨을 알 수 있다.

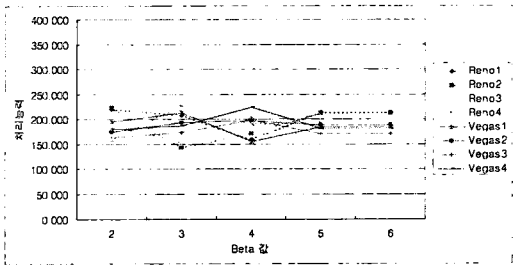
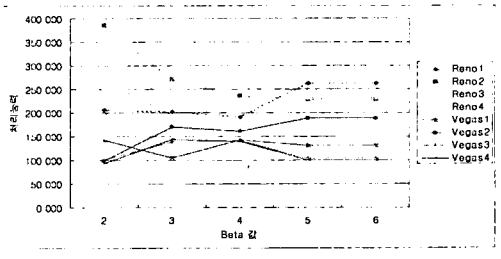
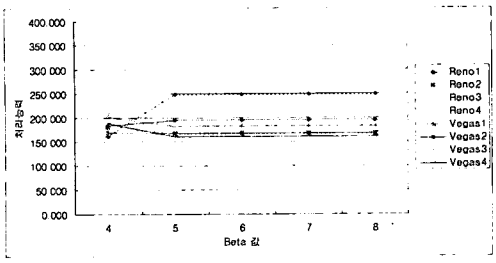


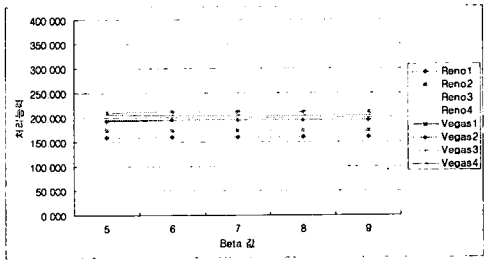
그림 2. α=1, delay = 1ms



(a) α = 1인 경우



(b) α = 3인 경우



(c) α = 4인 경우

그림 3. delay =80ms

두 버전간의 가장 좋은 공정성은 α=3, 4인 경우였으며, $\alpha \geq 4$ 이상인 경우 두 버전간의 공정성이 역전되어 그 이후로는 공정성이 떨어지는 모습을 보였다. α와 β의 차이가 2 이하인 경우 공정성이 가장 좋았으며, 앞의 경우와 같이 α와 β의 차이가 4이상인 경우 α와 β 차이에 의한 공정성의 변화는 일어나지 않았다. 또한, α가 증가함에 따라 공정성에 변화를 일으키는 α와 β의 차이는 점점 작아짐을 알 수 있다.

TCP Vegas와 TCP Reno의 공정성이 좋을 때의 TCP Vegas의 cwnd값을 보여주는 그림 4를 관찰해 보면 보다 큰 cwnd 값을 가지기 위해 cwnd 값의 변동이 커지는 것을 알 수 있다.

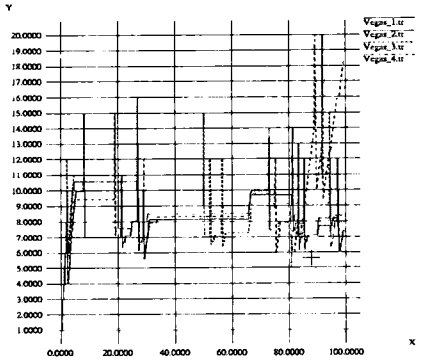


그림 4. α=3, β=4일 때 Vegas의 cwnd

이에 따라, TCP Vegas의 식에 의해서 α,β 값이 커질 경우 최대 cwnd 값이 커지게 되고 라우터 버퍼에 보다 많은 패킷이 머무르게 되고 이에 따라 TCP Reno와의 공정성이 향상된다는 것을 시뮬레이션을 통해 확인할 수 있다.

4. 결 론

네트워크의 delay가 적은 경우 α,β 값이 작은 경우에는 TCP Vegas와 TCP Reno의 공정성이 높았으나, 네트워크의 delay가 커질 경우 두 버전간의 공정성이 나빠지는 것을 확인할 수 있었다. 이에 α, β값을 증가시킬 경우 TCP Vegas와 TCP Reno의 공정성을 확보할 수 있었으며, 이 경우 TCP Vegas의 cwnd 값이 증가하는 모습을 볼 수 있다. 두 버전간의 공정성을 확보하기 위한 최적의 α, β 값은 네트워크의 delay에 따라 달라지기 때문에 α, β를 고정적인 상수값이 아닌 네트워크의 상황에 따른 가변적인 값으로 주어질 경우 두 버전간의 공정성 문제를 해결할 수 있다는 사실을 확인하였다. 이에 향후 가변적인 α,β값을 구하기 위한 계속적인 연구가 필요하다.

5. 참고문헌

[1] L. S. Brakmo and L. L .Peterson, "TCP Vegas: End to and congestion avoidance ona global Internet," IEEE J. Select. Areas Commun., vol. 13, pp.1465-1480, Oct. 1995.

[2] J. Mo, R. J. La, V. Anantharam, and J. Walrand, "Analysis and Comparison of TCp Reno and Vegas," IEEE INFOCOM '99, pp.1556-1563, 1999.

[3] O. Ait-Hellal and E. Altman, "Analysis of TCP Vegas and TCP Reno," IEEE ICC '97, pp.495-499, 1997.

[4] G. Hasegawa, K, Kurata and M, Murata. "Analysis and improvement of fairness between TCP Reno and Vegas for deployment of TCP Vegas to the Internet," IEEE Network Protocols, 2000. Proceedings. 2000 International Conference on , 2000, pp.177-186, 2000.

[5] D.D. Luong and J. Biro, "On the proportional fairness of TCP vegas," Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE, Vol. 3 , pp.: 1718-1722, 2001