

BBRv2 혼잡제어 알고리즘의 프로토콜 간 공정성 비교 실험

서상진*, 송영준, 김건환, 조유제

경북대학교 전자공학부

{tjtkdwls1111, syj5385}@knu.ac.kr, {kgh76, yzcho}@ee.knu.ac.kr

Comparative Experiment for Inter-protocol Fairness of BBRv2 Congestion Control Algorithm

Sang-Jin Seo*, Yeong-Jun Song, Geon-Hwan Kim, and You-Ze Cho

School of Electronics Engineering Kyungpook National University

요약

2016년 구글이 제안한 새로운 혼잡제어 알고리즘인 Bottleneck Bandwidth Round-trip propagation time (BBR)은 손실기반 혼잡제어 알고리즘과 동일한 링크에서 공존하는 경우에 두 흐름의 처리량은 병목 버퍼의 크기에 의존한다. 따라서, 구글은 이러한 문제점을 해결하기 위하여 BBRv2를 개발하고 있다. 하지만 초기에 공개된 BBRv2- α 버전과 손실기반 혼잡제어 알고리즘이 2 BDP 이상의 충분한 크기의 버퍼에서 동작할 때 두 알고리즘 간의 처리량 불균형은 이전 BBRv1과 유사했다. 본 논문에서는 BBRv1과 BBRv2- α 버전의 병목 버퍼 크기에 따른 알고리즘 간 공정성을 미니넷 기반의 실험 환경을 통해 실험 및 평가하였다.

I. 서론

2016년 구글이 제안한 Bottleneck Bandwidth Round-trip propagation time (BBRv1)은 주기적으로 최대 대역폭의 크기와 최소 지연 시간을 측정하여 최대 전송 속도 및 낮은 지연 시간을 보장한다 [1]. 그러나, BBRv1과 CUBIC이 같은 링크를 공유할 때 두 알고리즘의 서로 다른 동작 특성으로 인해 2 BDP 미만의 병목 버퍼에서 성능 불균형의 문제점과 2 BDP 이상의 병목 버퍼에서 심각한 성능 변화에 대한 문제가 제기되었다 [2], [3].

구글은 BBRv1의 동작 이슈를 개선하기 위하여 테스트 버전인 BBRv2- α 를 새로 공개하였다 [4]. 2 BDP 미만의 병목 버퍼 환경에서 BBRv2와 CUBIC 흐름이 동일한 링크를 공유하는 경우 BBRv1보다 알고리즘 간 공정성이 개선되었지만, 2 BDP 이상의 병목 버퍼를 가진 링크에서의 두 알고리즘 간의 데이터 전송은 여전히 공정하지 않았다 [5].

본 논문에서는 미니넷 에뮬레이터를 통해 BBRv1과 BBRv2 혼잡제어 알고리즘이 CUBIC과 공통의 링크에서 공존할 때, 병목 버퍼의 크기에 따른 알고리즘 간 공정성을 평가하였다.

II. 본론

A. 시뮬레이션 구조



그림 1. 미니넷 실험 환경 구성도

그림 1은 서로 다른 알고리즘을 이용하는 두 개의 흐름이 동일한 링크를 공유하는 실험 환경을 나타낸다. 호스트 1은 실험 시나리오에 따라

BBRv1 또는 BBRv2 혼잡제어 알고리즘을 이용하며, 호스트 2는 CUBIC 혼잡제어 알고리즘을 이용한다. 스위치 1은 *ifb* 과 *tc*를 이용하여 50Mbps의 병목 링크, 30ms의 왕복 지연 시간 및 0.2 ~ 32 BDP 크기의 병목 버퍼를 설정하였다. 두 개의 호스트는 *iperf* 응용프로그램을 이용하여 100초 동안 동시에 데이터를 전송한다.

B. 실험결과

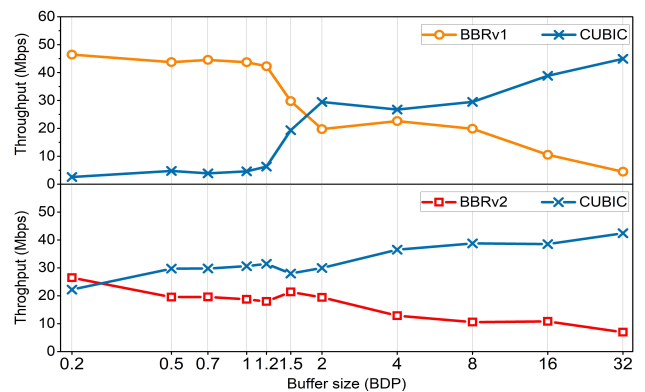
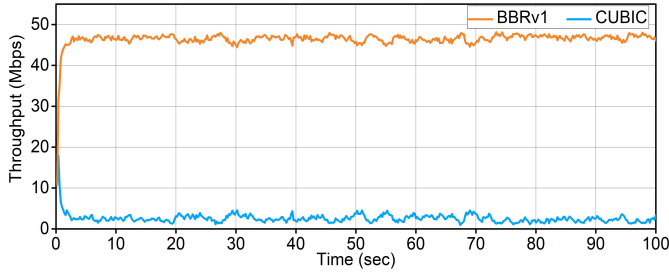


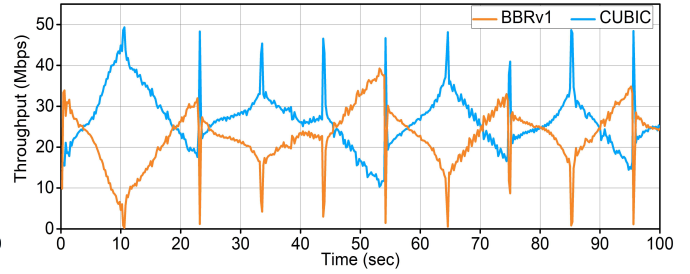
그림 2. 병목 버퍼 크기에 따른 BBRv1, BBRv2와 CUBIC의 처리량

그림 2는 BBRv1 및 BBRv2가 각각 CUBIC과 공통의 링크에서 공존할 때 병목 버퍼의 크기에 따른 처리량을 나타낸다. 병목 버퍼의 크기가 2 BDP보다 작은 경우 BBRv1 흐름은 대부분의 대역폭을 차지하여 CUBIC 흐름을 완전히 압도하는 반면, BBRv2의 처리량은 CUBIC과 유사한 것을 확인할 수 있다. 하지만 2 BDP 이상의 병목 버퍼에서 동작하는 경우에는 기존의 BBRv1과 유사하게 버퍼의 크기가 증가할수록 점차 CUBIC의 처리량이 증가하는 모습을 보인다.

그림 3(a)는 0.2 BDP의 병목 버퍼에서 BBRv1과 CUBIC의 시간에 따른 처리량을 나타내며 BBRv1이 CUBIC을 압도해 대부분의 대역폭을 점유하는 것을 확인할 수 있다. BBRv1은 최대 2 BDP까지 데이터를 전송하

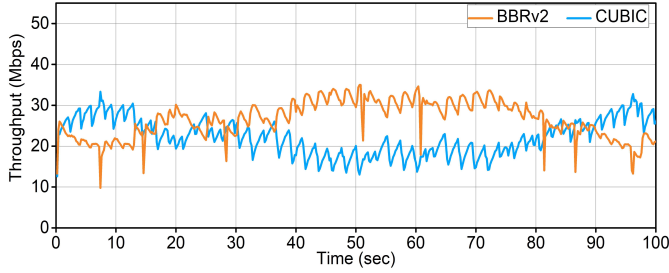


(a) 0.2 BDP의 병목 버퍼

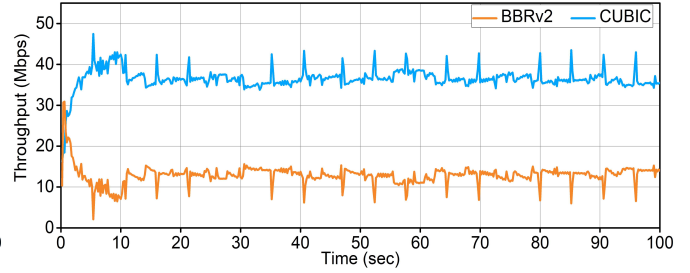


(b) 4 BDP의 병목 버퍼

그림 3. BBRv1 vs CUBIC의 시간에 따른 처리량 비교



(a) 0.2 BDP의 병목 버퍼



(b) 4 BDP의 병목 버퍼

그림 4. BBRv2 vs CUBIC의 시간에 따른 처리량 비교

기 때문에 0.2 BDP의 병목 버퍼에서는 버퍼 범람으로 인한 패킷손실이 발생한다. 이때의 패킷손실을 네트워크 혼잡 신호로 인식하지 않는 BBRv1은 전송량을 줄이지 않지만, CUBIC은 손실에 따라 혼잡윈도우의 크기를 줄이기 때문에 BBRv1이 우세한 성능을 보인다.

그림 3.(b)는 4 BDP의 병목 버퍼에서 BBRv1과 CUBIC의 처리량을 나타낸다. BBRv1은 ProbeRTT 동작에서 CUBIC의 특성으로 인해 실제로 다 큰 최소 RTT (minRTT)를 측정하여 많은 양의 데이터를 전송한다. 반면 CUBIC은 slow start로 버퍼를 가득 채울 때까지 전송량이 증가하는 특성을 가진다. BBRv1은 ProbeRTT 동작 이후 점유했던 버퍼를 비우며 실제와 가까운 minRTT를 측정하기 때문에 처리량이 감소하고, 이로 인해 CUBIC이 대부분의 버퍼를 점유하게 된다. 이 동작이 반복되어 주기적인 성능 변동이 발생한다.

0.2 BDP의 병목 버퍼에서 BBRv2와 CUBIC의 처리량을 나타내는 그림 4.(a)를 통해 BBRv2와 CUBIC 흐름이 공존하는 경우 두 알고리즘의 처리량은 기존 BBRv1보다 공정성이 개선된 것을 확인할 수 있다. BBRv2는 BBRv1과 달리 임계 손실률을 초과하는 패킷손실이 발생하는 경우 전송률을 줄여 마지 손실기반 혼잡제어 알고리즘과 유사하게 동작하기 때문에 CUBIC과의 공정성이 향상되었다.

그림 4.(b)는 4 BDP의 병목 버퍼에서 BBRv2와 CUBIC의 처리량을 나타낸다. BBRv2는 대역폭과 minRTT를 측정해 만든 모델을 기반으로 전송량을 1 BDP로 제한하는 반면 CUBIC은 패킷손실이 발생할 때까지 전송량을 증가시킨다. BBRv2가 내보내는 데이터의 양이 CUBIC에 비해 적기 때문에 상대적으로 CUBIC이 우세한 성능을 보인다.

III. 결론

본 논문에서는 BBRv2- α 버전이 손실기반 혼잡제어 알고리즘을 이용하는 흐름과 동일한 링크를 공유할 때의 알고리즘 간 공정성을 비교 및 분석하기 위해 미니넷 에뮬레이터 기반의 실험 환경을 구성하였다. 병목 버퍼의 크기가 2 BDP 미만일 때 BBRv2의 알고리즘 간 공정성은 개선되었으나 2 BDP 이상의 버퍼를 가진 링크에서는 여전히 두 알고리즘 간 불

공정이 존재함을 확인하였다.

ACKNOWLEDGMENT

This research was supported by Next Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (No. NRF-2017M3C4A7083676) and by Basic Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1A6A1A03025109) and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2019R1A2C1006249).

참고 문헌

- [1] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-based congestion control," ACM Queue, vol. 14, no. 5, pp. 50:20–50:53, 2016.
- [2] 김진환, 송영준, 박창훈, 조유제, "BBR 혼잡제어 알고리즘 표준화 및 연구 동향 분석," 한국통신학회논문지, 제 44권, 제 9호, pp. 1713–1722, 2019.
- [3] M. Hock, R. Bless, and M. Zitterbart, "Experimental evaluation of BBR congestion control," in Proc. International Conference on Network Protocols (ICNP), 2017.
- [4] N. Cardwell, Y. Cheng, S. H. Yeganeh, I. Swett, V. Vasiliev, P. Jha, Y. Seung, M. Mathis, and V. Jacobson, "BBRv2: A Model-based Congestion Control," in Proc. IETF 104 meeting, 2019. [Online]. Available: <https://datatracker.ietf.org/meeting/104/materials/slides-104-iccr-g-an-update-on-bbr-00>.
- [5] J. Gomez, E. Kfoury, J. Crichigno, and E. Bao-Harb, and G. Srivastava, "A Performance Evaluation of TCP BBRv2 Alpha," in Proc. International Conference on Telecommunications and Signal Processing, 2020.