

# 원거리 망에서 MTU 크기 및 병렬화에 따른 BBRv3 성능 측정 및 분석

홍원택

한국과학기술정보연구원 과학기술연구망센터

wthong@kisti.re.kr

## Measurement and Analysis of BBRv3 Performance for MTU size and Parallelism in Wide Area Network

Wontaek Hong

KREONET Center, KISTI

요 약

Google에서 제안된 BBR(Bottleneck Bandwidth and RTT)은 기존 손실기반 혼잡제어 알고리즘의 동작 방식과 달리 최대 대역폭과 최소 지연을 지향하는 Network path 모델을 기초로 동작한다. 현재까지 BBRv1, v2의 단점들을 해결한 BBRv3가 공개되어 특정 버전의 리눅스 커널에서 적용 및 실험이 가능하다. 본 논문에서는 BBRv3를 대상으로 약 110ms에 달하는 한미간 원거리 망에서 패킷 손실률, Maximum Transmission Unit(MTU) 크기, 전송 스트림들의 병렬화 정도의 조합에 따른 중단간 전송 성능을 측정하고, 기존 손실기반 혼잡제어 알고리즘인 Cubic의 처리율과 비교 분석을 수행한다.

시한다.

### 1. 서론

대용량의 과학응용 데이터의 고속 전송 및 공유를 지원하기 위한 ScienceDMZ 모델에 따르면 고성능 네트워크, 전용의 데이터 전송 노드(DTN), 망 성능 측정 노드 및 접근 제어 목록의 관리 등이 필요하다 [1]. 이러한 환경에서 이용되는 전용의 DTN에는 GridFTP/Globus와 같은 데이터 전송 툴이 탑재되고, 많은 전송 툴들은 TCP 전송 제어 프로토콜에 의존적이다. 특히 TCP 알고리즘과 관련하여 Cubic, Htcp과 같은 손실기반 혼잡제어 알고리즘(CCA)은 여전히 가장 많이 활용되고 있고, 최근에는 Google에서 제안된 BBR(Bottleneck Bandwidth and RTT)이 또 다른 선택지로서 검토되고 있다.

본 논문에서는 중단간 RTT가 약 110ms에 달하는 한미간 원거리 네트워크 환경에서 가상머신과 베어메탈 서버를 이용하여 BBRv3 테스트베드를 구성하고, ScienceDMZ 기반 대용량 데이터의 고속 전송 시 전송 성능에 영향을 줄 수 있는 성능 메트릭들 중 패킷 손실률, MTU 크기, 전송 스트림들의 병렬화 정도에 따른 BBRv3의 처리율을 측정하고 손실기반 CCA인 Cubic의 처리율과 비교하여 결과를 제

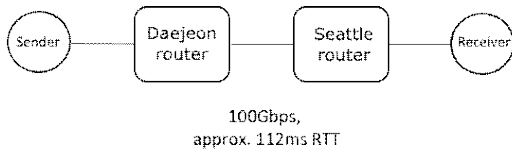
### 2. 본론

Ubuntu, Debian 등의 리눅스 최신 배포판에는 이미 BBRv1이 탑재되어 있지만, 다른 CCA들의 플로우들과의 공평성 문제, shallow buffer에서의 높은 재전송률, RTT가 긴 BBR 플로우들과의 공평성 문제 등으로 인해 실제 망에서 적용하기에는 문제점이 존재하고, 이러한 문제점들을 보완한 후속 버전들이 발표되어 왔다. 특히, BBRv3는 언급된 문제들의 해결을 포함하여 Bandwidth proving 단계에서의 지연된 Bandwidth convergence 이슈, Startup 단계에서의 파라미터 튜닝 등을 통한 성능 향상을 추구하였고 [2], 현재 [3]에서 공개되어 특정 버전의 리눅스 커널을 통해 적용 및 실험해 볼 수 있다.

TCP 기반으로 동작하는 망에서의 처리율과 관련하여 Mathis equation에 따르면 전송 성능은 Maximum Segment Size(MSS), RTT, 패킷 손실률에 영향을 받게 되어 있고, TCP 기반의 GridFTP/Globus와 같은 전송 프로토콜 및 어플리케이션에서 처리율은 전송 스트림들의 병행성, 병렬성 정도에 따라 성능 향상을 기대할 수 있다 [1][4]. 이

와 관련하여 본 논문에서는 손실기반 CCA인 Cubic 대비 BBRv3의 전송 성능을 측정 및 비교 분석하기 위해 고정된 RTT를 기준으로 패킷 손실률, MTU 크기, 전송 스트림들의 병렬화 정도를 성능 메트릭으로 선정하여 전송 실험을 수행한다.

종단간 전송 실험이 수행된 실험 환경은 다음과 같다. 송신측 종단포인트는 AMD EPYC 3.2GHz의 CPU, 32GB RAM, 100Gbps 물리 NIC에서 가상화된 네트워크 인터페이스를 갖는 가상머신으로 구성되고 22.04 버전의 Ubuntu 리눅스를 설치한 후, 6.4.0+v3+7542cc7c41c0+GCE 버전의 BBRv3 커널 컴파일 및 커널 이미지를 로딩하여 적용한다. 또한, 일종의 vDTN(virtual DTN) 관점에서 TCP 소켓 버퍼를 약 110ms에 달하는 한미간 RTT를 반영하는 Large BDP(Bandwidth Delay Product)로 튜닝한다. 수신측 종단포인트는 ScienceDMZ 모델에서 성능 측정용으로 이용되는 iperf3 서버를 활용하고, 해당 서버는 Intel Xeon 3.5GHz의 CPU, 256GB RAM, 100Gbps 물리 NIC으로 구성된다. 송신 종단포인트와 수신 종단포인트는 종단간 병목 없는 100Gbps 망으로 구성되고, 기본 설정은 MTU9000을 반영하여 대용량 데이터 전송에 적합한 점보 프레임을 전송할 수 있는 환경으로 구성된다.

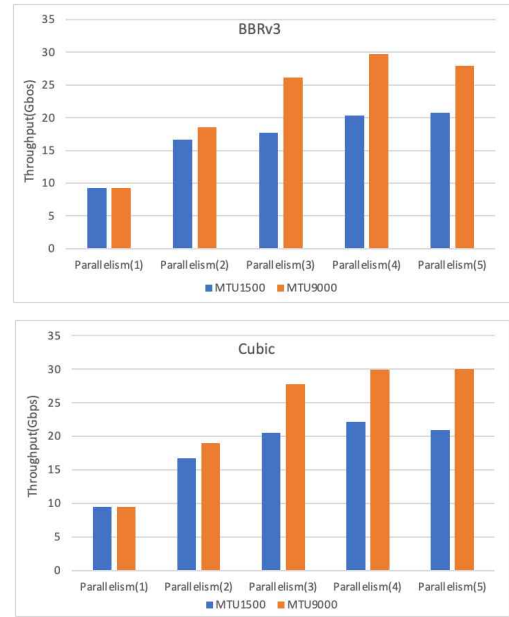


(그림 1) 실험 토폴로지

실험 구간은 패킷 손실이 없는 환경(Case I)과 NetEm을 이용해 0.01%의 패킷 손실률을 적용한(Case II) 두 가지 실험 케이스들로 분류하여 진행한다. 각 케이스별로 MTU1500, MTU9000에 전송 스트림들의 수를 1에서 5까지 변화시킴으로써 병렬화 정도에 변화를 주면서 BBRv3와 Cubic에 대한 iperf3 전송 실험을 120초간 수행하고, 동일한 실험을 3회 반복하여 평균값을 산출한다.

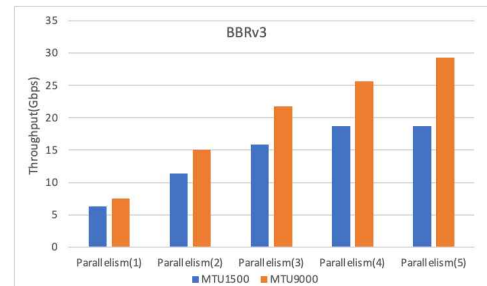
위 실험에서 수행된 Case I, Case II에 대하여 Cubic, BBRv3의 처리율을 비교한다. Case I에서는 Cubic, BBRv3 모두에 대해 MTU 크기 및 전송 스트림들의 병렬화 정도에 따른 처리율 이득을 확인할 수 있다. 전송 스트림들의 병렬화에 따른 이득은 P=4에서 정점을 기록하였고 두 CCA에 대해 모든

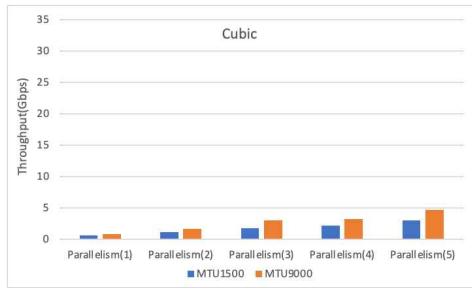
실험 케이스들에서 유사한 처리율을 보였다.



(그림 2) Throughput - Case I

Case II에서는 Case I과 같이 두 CCA에 대해 MTU 크기 및 전송 스트림들의 병렬화 정도에 따른 처리율 이득은 지속된다. 단, Cubic의 경우 손실기반 혼잡제어 메커니즘에 따라 동작하므로 0.01%의 패킷 손실률에도 병렬화 P=5에서 약 3~5Gbps의 상대적으로 낮은 처리율을 기록한다. 이에 비해 BBRv3의 경우 MTU1500, MTU9000에 대해 병렬화 P=5 기준으로 약 19Gbps, 29Gbps의 처리율을 각각 산출한다. 이러한 성능 차이는 단순히 패킷 손실을 혼잡으로 인식하는 Cubic의 손실기반 혼잡제어 방식과 달리, BBRv3에서는 Bufferbloat 문제를 해결하면서 네트워크 버퍼와 경로 상에서의 “실질적인” 혼잡을 고려하는 Network path 모델을 기반으로 동작하기 때문으로 추정된다.





(그림 3) Throughput - Case II

위의 실험 결과 분석과 관련하여 iperf3의 동작 메커니즘에서 단일 코어 CPU 상에서 제한적으로 전송 스트림들의 병렬화가 적용되는 점, BBRv3가 적용된 가상 머신이 속한 KVM 하이퍼바이저 상에서의 NIC 최대 대역폭 상한, 한미간 구간에서 본 실험 외에 실제 전송되는 Production traffic 등의 제약 요소들은 본 실험에서 산출된 처리율에 영향을 줄 수 있는 부분이므로 결과 해석에 적절히 고려되어야 한다.

### 3. 결론

본 논문에서는 Google에서 개발되어 공개된 BBRv3 커널 소스를 컴파일한 후 커널 이미지를 가상머신에 탑재하고 한미간 원거리 망에서 패킷 손실률, MTU 크기, 전송 스트림들의 병렬화 정도에 따른 조합을 기반으로 중단간 처리율을 측정하고, 기존 손실기반 CCA인 Cubic의 처리율과 비교하였다. 실험 결과 0.01%의 패킷 손실률에서 전송 스트림들에 대한 병렬화 P=5를 기준으로 MTU9000에 대해 약 6.2배의 처리율 증가를 가져오는 것을 확인할 수 있었다. 이러한 결과는 0.01% 정도의 패킷 손실률이 존재하는 네트워크 환경에서 기존 손실기반 CCA인 Cubic은 현저한 처리율의 저하를 가져오는 반면, 패킷 손실에 덜 민감한 BBRv3의 경우 여전히 MTU 크기, 전송 스트림들의 병렬화에 따른 성능 이득을 가져올 수 있음을 의미한다.

### ACKNOWLEDGMENT

이 논문은 2025년도 한국과학기술정보연구원 (KISTI)의 기본사업으로 수행된 연구입니다.(과제번호: (KISTI)K25L5M1C1)

### 참고문헌

[1] J. Crichigno, E. Bou-Harb, and N. Ghani, "A Comprehensive Tutorial on Science DMZ", IEEE

Communications Surveys & Tutorials, vol. 21, no. 2, pp. 2041-2078, 2019.

[2] N. Cardwell et al., "BBRv3: Algorithm Bug Fixes and Public Internet Deployment," Proceedings of IETF 117th meeting, 2023.

[3] BBRv3 [Internet]. Available : <https://github.com/google/bbr/blob/v3/README.md>

[4] W. Hong et al., "Deployment and Performance Analysis of Data Transfer Node Cluster for HPC Environment," KIPS Transactions on Computer and Communication Systems, vol. 9, no. 9, pp. 197-206, 2020.