

TCP 성능 향상을 위한 새로운 상향링크 혼잡 제어 기법

손 경 호[°], 김 한 석^{*}, 곽 동 호^{*},
 아브셰크 로이^{*}, 김 동 숙^{*}, 김 영 용^{**}

Novel Uplink Congestion Control Method for TCP Throughput Enhancement

Kyungho Sohn[°], Han-Seok Kim^{*},
 Dongho Kwak^{*}, Abhishek Roy^{*},
 Dongsook Kim^{*}, Young Yong Kim^{**}

요 약

본 논문은 TCP 수정 없이 상향링크 혼잡 상황에서 확인응답 지연시간 개선을 통해 하향링크 TCP 성능을 향상시킬 수 있는 새로운 상향링크 혼잡 제어 기술을 제안한다. 제안한 방법으로 하향링크 TCP 성능이 향상됨을 실험을 통해 확인한다.

Key Words : TCP, Congestion Control, Throughput, TCP ACK, LTE

ABSTRACT

In this paper, we propose a novel uplink congestion control scheme, which enhances downlink TCP throughput by improving response time of TCP acknowledgements without TCP modification. Through the experimental results, it is manifested that the proposed scheme is able to achieve better downlink TCP throughput.

I. 서 론

TCP(Transmission Control Protocol)는 데이터 전송의 신뢰성 보장을 위해 데이터를 받은 수신단에서

확인응답(Acknowledgement)을 전송한다^[1,2]. 따라서 하향링크 데이터 전송에 대한 확인응답을 위해 일정량의 상향링크 자원을 사용해야만 하는데, 상향링크 혼잡 상황에서는 우선순위가 높은 음성 혹은 비디오 등의 보장형 비트레이트(Guaranteed Bit Rate) 서비스 및 상향링크 TCP 처리율을 높이기 위해 확인응답에 대한 지연이 발생함으로써 하향링크 자원에 여유가 있음에도 불구하고 하향링크 TCP 처리율이 저하되는 현상이 발생하게 된다. 특히 이 문제는 상향링크 자원이 하향링크 자원에 비해 부족한 시분할 LTE(Time-Division LTE) 시스템^[3]이나 하향 반송과 묶음(Carrier Aggregation) 기법^[4]만 지원하고 상향 반송과 묶음 기법은 지원하지 않는 경우에 심각하게 발생할 수 있으나, 현재까지 이와 관련된 연구가 진행된 적이 없다. 따라서 본 논문에서는 상향링크 혼잡 상황에서 하향링크 TCP 처리율의 저하현상에 대한 원인이 확인응답 지연임을 최초로 확인하고, TCP 수정 없이 하향링크 TCP 처리율을 향상시키기 위한 상향링크 혼잡 제어 기법을 제안한다. 제안하는 기법은 TCP 정보 없이 TCP 혼잡 제어를 유도하며, 파라미터 설정을 통해 단순 drop 방식 대비 더욱 효과적으로 혼잡 제어를 수행할 수 있는 장점이 있다.

II. 제안하는 상향링크 혼잡 제어 기법

본 장에서는 먼저 본 기법을 제안하게 된 동기에 대해 설명하고, 이 문제를 해결하기 위해 상향링크 혼잡 상황에서 상향링크의 자원이 하향링크 확인응답에 효율적으로 이용될 수 있도록 상향링크 TCP 트래픽을 제어하는 방법을 제안한다.

2.1 기법을 제안한 동기

그림 1은 상향 및 하향링크 자원 상황에 대한 예시이다. U 는 상향링크 TCP 트래픽 양이며, A 는 하향링크 TCP 트래픽을 위한 확인응답 양이다. D_{GBR} 및 U_{GBR} 은 각각 상향 및 하향링크 보장형 비트레이트 트래픽 양이다. 이때 σ 를 TCP 트래픽 양과 확인응답 양의 비율이라고 가정하면, 하향링크 최대 처리율은 $D_{GBR} + U \times \sigma + A/\sigma$ 로 제약된다. 기존 스케줄러는 상향링크 처리율을 높이기 위해 U_{GBR} 를 보장하고 U 와 A 의 우선순위를 구별하지 않기 때문에 상향링크

[°] First and Corresponding Author : Yonsei University School of Electrical and Electronic Engineering, heroson7@gmail.com, 경희원
^{*} Samsung Electronics, Networks Business, {(hs365.kim, dongho1.kwak, abhishek.roy, kimds.kim}@samsung.co.kr}, 정희원
^{**} Yonsei University School of Electrical and Electronic Engineering, y2k@yonsei.ac.kr, 종신희원
 논문번호 : KICS2016-10-320, Received October 20, 2016; Revised November 17, 2016; Accepted December 20, 2016

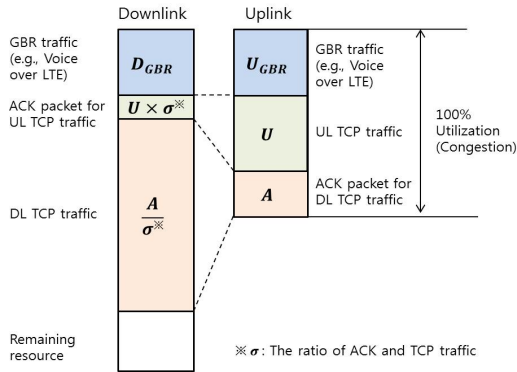


그림 1. 비대칭 LTE 시스템에서 상향 및 하향 자원 할당
Fig. 1. Resource assignment for uplink and downlink in asymmetric LTE systems

혼잡 상황에서는 A 의 전송이 지연될 수 있다. 이 때문에 하향링크에서 A/σ 가 제한됨으로써 하향링크 최대 처리율이 저하되었다. 따라서 이를 향상시키기 위해서 A 의 전송지연을 줄여야하며, U_{GBR} 은 보장형 비트레이트이므로 기지국에서 조절할 수 없기 때문에 혼잡 상황에서는 U 를 조절하여 A 의 전송시간을 개선하는 상향링크 혼잡 제어 기법의 개발이 필요하다.

2.2 상향링크 혼잡 제어 기법

본 장에서는 하향링크 TCP 성능향상을 위한 상향링크 혼잡 제어 기법을 제안한다. 제안하는 기법은 TCP 수정 없이 기지국이 이미 알고 있는 셀 내 단말 수 및 단말의 BSR(buffer status report) 정보를 이용하여 TCP 혼잡 제어를 유도하므로 TCP 정보 없이 동작하는 장점이 있다. 상향링크 혼잡 제어 기법은 매 상향링크 TTI(Transmission Time Interval) 마다 다음의 절차로 진행된다.

- (1) (혼잡 제어 판단) 셀 내 모든 단말의 BSR을 이용하여 이 합계가 기준치 TH_Cell 보다 크면 혼잡 제어를 시행한다.
- (2) (대상 단말 결정) 해당 TTI에 스케줄링되어 자원을 할당받은 단말 중 버퍼량이 기준치 TH_UE 보다 큰 단말 하나를 임의로 선택한다.
- (3) (혼잡 제어 결정) 선택된 단말에 대해 p 의 확률로 혼잡 제어 수행여부를 결정한다.
- (4) (혼잡 제어 실행) 혼잡 제어 수행 대상 단말에서 기준치 TH_Packet 보다 큰 패킷이 수신되면 상위로 전송하지 않고 기지국 내부에서 제거하여 혼잡 제어를 실행한다.

2.3 효율적인 혼잡 제어를 위한 파라미터 설정

본 장에서는 효율적인 혼잡 제어를 위해 시스템 사용자가 설정해야할 파라미터에 대해 설명한다. 제안하는 방식은 이들 설정 파라미터를 잘 조절하여 기존의 단순 drop 방식 대비 효과적인 TCP 혼잡 제어를 수행할 수 있는 장점이 있다.

2.3.1 혼잡 제어 판단 기준치 TH_Cell 설정

TH_Cell 설정을 위한 설계 개념은 아래와 같다.

- (i) 한 단말의 최대 처리율은 보장한다. 따라서 셀 내 한 단말이 있을 때 TH_Cell_1 는 대역폭-지연 곱 (bandwidth-delay product)^[5]에 의해 식 (1)과 같다.

$$TH_Cell_1 = (4/3) \times T_1 \times RTT_{target} \quad (1)$$

여기서 T_1 및 RTT_{target} 은 단말이 한 대일 때, 셀 최대 처리율(Peak Throughput) 및 목표 왕복 시간(Round Trip Time)이다. 계수 4/3은 TCP 혼잡 제어 동작으로 혼잡 윈도우 크기가 절반으로 떨어지더라도 최대 처리율을 보장하기 위해 곱해진다. (ii) 셀 혼잡 기준을 정하여 그 이상이 될 경우 항상 동작한다. 이는 상향링크 TCP 트래픽(U)과 하향링크 TCP 트래픽을 위한 확인응답(A)이 상향링크 자원사용 측면에서 서로 경쟁관계에 있으므로 A 를 위한 상향링크 자원을 확보하기 위해 U 가 과도하게 커지는 것을 제어하기 위함이다. 따라서 혼잡의 기준이 되는 단말수를 N , 이때 해당 단말들에 혼잡 제어를 동작시키기 원하는 목표 혼잡 윈도우를 CW_{target} 이라하면 혼잡 기준일 때 TH_Cell_N 은 식 (2) 와 같다.

$$TH_Cell_N = N \times CW_{target} \times MSS \quad (2)$$

여기서 MSS 는 최대 세그먼트 크기(Maximum Segment Size)이다. 참고로 본 논문에서는 항상 MSS 로 패킷을 전송한다고 가정한다. (iii) 혼잡 기준 이하의 단말수에 대해서는 TH_Cell_1 과 TH_Cell_N 간의 선형보간법(Linear Interpolation)^[7]을 이용하여 결정한다. 식 (1), (2) 및 이를 정리하면 셀 내에 n 대의 단말이 있을 때, TH_Cell_n 는 식 (3)과 같다.

$$TH_Cell_n = \begin{cases} (4/3) \times T_1 \times RTT_{target}, & n = 1 \\ (n-1) TH_Cell_N \\ + \frac{N-1}{N-1} TH_Cell_1, & 1 < n < N \\ N \times CW_{target} \times MSS, & n \geq N \end{cases} \quad (3)$$

2.3.2 대상 단말 기준치 TH_{UE} 설정

혼잡 제어를 받게 되면 TCP 혼잡 윈도우의 크기가 절반으로 떨어지게 되므로 CW_{target} 의 절반보다 큰 BSR을 가지는 단말을 선택하는 것이 타당하다. 따라서 TH_{UE} 는 식 (4)와 같다.

$$TH_{UE} = (CW_{target}/2) \times MSS \quad (4)$$

2.3.3 혼잡 제어 결정 확률 p 설정

혼잡 제어를 수행하면 TCP 혼잡 윈도우가 감소하므로 셀의 혼잡 수준이 감소하고, 시간이 지나면 TCP 혼잡 윈도우가 다시 증가하므로 셀의 혼잡 수준도 다시 증가한다. 따라서 적절한 혼잡 제어 확률 p 를 설정함으로써 셀의 혼잡 제어 수준을 정상 상태(steady state)로 유지할 수 있다. 즉 p 는 셀 내의 단말수 및 CW_{target} 까지 걸리는 TTI 수와 관련이 있고 이는 식 (5)와 같다.

$$p = n / (\tau \times M) \quad (5)$$

여기서 n 는 단말수, τ 는 혼잡 제어로 인해 감소된 TCP 혼잡 윈도우가 CW_{target} 으로 복구되어 다시 혼잡 제어를 받을 때까지 걸리는 시간, 그리고 M 은 해당 시스템의 초당 상향 TTI 수이다. LTE는 시스템 구성에 따라 M 이 상이하므로 이를 반영한 것이다(예를 들어, TD-LTE U/D configuration 2는 $M=200$). τ 는 전송하고자 하는 패킷량 및 단말 당 최대 처리율과 관련이 있고 이는 식 (6)과 같다.

$$\begin{aligned} \tau &= \frac{MSS \times 8 \times \sum_{i=CW_{target}/2}^{CW_{target}-1} i}{T_{peak}/n} \\ &= \frac{MSS \times (3CW_{target}^2 - 2CW_{target})}{T_{peak}/n} \end{aligned} \quad (6)$$

여기서 T_{peak} 는 셀 최대 처리율이다. 식 (6)을 식 (5)에 적용하면 최종적으로 혼잡 제어 결정 확률 p 를 구할 수 있으며, 이는 식 (7)과 같다.

$$p = \frac{T_{peak}}{MSS \times (3CW_{target}^2 - 2CW_{target}) \times M} \quad (7)$$

또한 구현 편의를 위해 $CW_{target}/2$ 는 정수로 정한다.

2.3.4 혼잡 제어 실행 기준치 TH_{Packet} 설정

혼잡 제어를 실행 했을 때 상향링크 TCP 패킷을 제거해야하므로 확인응답 패킷 크기보다 큰 값으로 설정한다. (예를 들어, 100 byte)

이와 같이 제안하는 방식은 TH_{Cell} 설정을 통해 한 단말의 최대 처리율을 보장하면서 A 에 필요한 상향링크 자원을 확보하기 위해 U 가 과도하게 커지는 것을 제어하고, p 설정을 통해 셀의 혼잡 제어 수준을 정상 상태로 유지할 수 있으며, TH_{Packet} 설정을 통해 상향링크 TCP 패킷을 골라서 제거할 수 있으므로 단순 drop 방식 대비 효과적인 TCP 혼잡 제어 수행이 가능하다.

III. 성능 분석 및 결론

본 장에서는 20MHz BW를 가지는 TD-LTE (U/D Configuration 2) 시스템의 한 셀에 상향 및 하향링크 강진계에서 TCP를 사용하는 단말 각 10 대가 존재하고 이들이 상향링크 혼잡을 겪을 경우, OPNET^[7] 시뮬레이터를 이용하여 제안한 기법을 사용하지 않았을 때와 사용하였을 때의 TCP 성능을 확인한다. 자세한 시뮬레이션 환경은 표 1에 기재되어 있다.

그림 2에서 점선은 상향링크 혼잡 제어 기법을 사

표 1. 시험 파라미터
Table 1. Simulation parameters

| Item | Value |
|---------------|-----------|
| N | 50 |
| CW_{target} | 6 |
| MSS | 1460 Byte |
| M | 200 |

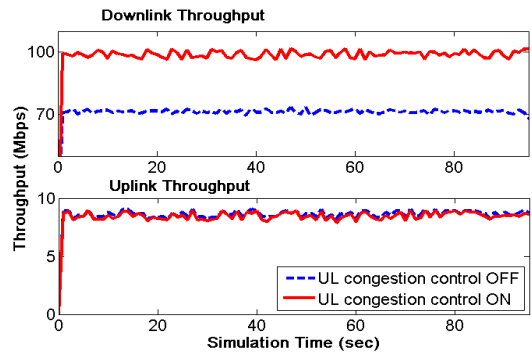


그림 2. 제안된 기법의 시험결과 ($p=0.031$)
Fig. 2. Experimental results of the proposed scheme ($p=0.031$)

용하지 않았을 때 측정된 TCP 처리율이며, 실선은 제안하는 상향링크 혼잡 제어 기법을 사용했을 때 측정된 TCP 처리율이다. 제안된 기법을 사용하였을 경우, 상향링크 TCP 처리율에 거의 영향을 주지 않으면서 확인응답 지연시간을 개선하여 월등히 향상된 하향링크 TCP 처리율을 얻을 수 있음을 알 수 있다. 일반적으로 확인응답 패킷의 크기가 TCP 패킷의 크기보다 매우 작기 때문에 약간의 상향링크 TCP 자원을 희생하면 큰 하향링크 자원을 확보할 수 있기 때문이다.

따라서 본 논문에서는 상향링크 혼잡 상황에서 하향링크 TCP 처리율의 저하현상에 대한 원인이 확인응답 지연임을 최초로 확인하고, 시험 결과를 통해 상향링크 자원이 하향링크 자원에 비해 부족한 TD-LTE 시스템에서 TCP 수정 없이 새롭게 제안한 상향링크 혼잡 제어 기법이 하향링크 TCP 성능을 향상시킬 수 있음을 확인하였다.

References

- [1] M. Allman, V. Paxson, and E. Blanton, *TCP congestion control*, RFC 5681, Sept. 2009.
- [2] W. Choi, Ramneek, and W. Seok, "Yellow-Light TCP: energy-saving protocol for mobile data transmission," *J. KICS*, vol. 40, no. 3, pp. 478-490, Mar. 2015.
- [3] 3GPP Technical Specification, 36.300, *E-UTRA and E-UTRAN Overall Description; Stage 2*, v14.0.0, 2016.
- [4] S. H. Lim and S.-W. Lee, "LTE-advanced CA features in 3GPP REL-12 and its future," *J. KICS*, vol. 39A, no. 9, pp. 497-507, Sept. 2014.
- [5] K. Chen, Y. Xue, S. H. Shah, and K. Nahrstedt, "Understanding bandwidth-delay product in mobile ad hoc networks," *Computer Commun.*, vol. 27, no. 10, pp. 923-934, 2014.
- [6] E. H. W. Meijering, "A chronology of interpolation: From ancient astronomy to modern signal and image processing," in *Proc. IEEE*, vol. 90, no. 3, pp. 319-342, Mar. 2002.
- [7] Riverbed Technologies, *OPNET Modeler*, from <https://www.riverbed.com/kr/index.html>.