

모바일 네트워크에서 TCP 성능 향상을 위한 패킷 포워딩 제어 방안

정회원 허 경*, 엄 두 섭*, 이승 현*, 차 균 현*

A Packet Forwarding Control Scheme for TCP Performance Improvement in Mobile Networks

Kyeong Hur*, Doo-Seop Eom*, Seung Hyun Lee*, Kyun Hyon Tchah* *Regular Members*

요 약

모바일 IP 프로토콜에서 핸드오프 동안에 발생하는 패킷 손실 때문에 초래되는 TCP의 성능저하를 방지하기 위해서는 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식에서 이전 기지국이 핸드오프 동안 손실된 패킷들을 버퍼에 저장하고 저장된 패킷들이 이동한 단말에게 전달되어야 한다. 그러나 이동단말이 혼잡한 라우터와 연결된 새로운 서브 네트워크로 이동한 경우에는, 이전 기지국이 포워딩하는 패킷들은 손실되고 이 패킷들로 인해 심화된 혼잡으로 인해 기존 고정 TCP 플로들의 링크이용률 성능이 저하되게 된다. 본 논문에서는 패킷 버퍼링 방식이 결합된 스무스 핸드오프 방식으로 이동단말이 혼잡이 존재하는 라우터와 연결된 새로운 서브네트워크로 이동한 경우, 새로운 서브 네트워크와 연결된 라우터가 사용하는 RED 버퍼 관리 방식의 혼잡 상태에 따라 이전 기지국이 저장한 패킷들을 포워딩하거나 폐기하는 패킷 포워딩 제어 방안을 제안하였다. 시뮬레이션 결과는 제안하는 패킷 포워딩 제어방안을 적용하여 링크이용률 성능을 향상시킬 수 있음을 보인다.

ABSTRACT

To prevent the performance degradation of TCP due to packet loss in the smooth handoff by the route optimization extension of Mobile IP protocol, a buffering of packets at a base station is needed. A buffering of packets at a base station recovers the packets dropped during the handoff by forwarding the buffered packets at the old base station to the mobile user. But, when the mobile user moves to a new foreign network which is connected to a congested router, the buffered packets forwarded by the old base station are dropped and the link utilization performance degraded due to increased congestion by the forwarded packets. In this paper, when the mobile user moves to a new foreign network which is connected to a congested router, we propose a packet forwarding control scheme required for the old base station to improve TCP performance in mobile networks. The old base station forwards or discards the buffered packets during handoff by proposed packet forwarding control scheme based on congestion states of RED(Random Early Detection) at the congested router. Simulation results show that link utilization performance can be improved by applying proposed packet forwarding control scheme.

1. 서 론

최근 수년간에 걸친 무선기술의 급속한 발전 및

이동단말의 폭발적 증가로 말미암아 사용자가 자유롭게 이동하면서 통신하는 것이 가능하게 되었다. 인터넷 환경에서도 이러한 사용자의 이동성을 지원

* 고려대학교 전자공학과 차세대 이동통신연구실(hkyeong@korea.ac.kr)

논문번호 : 020090-0225, 접수일자: 2002년 2월 25일

* "이 논문은 2000년도 한국학술진흥재단의 지원에 의하여 연구되었음" (KRF-2000-041-E00268)

하기 위하여, IETF(Internet Engineering Task Force)에서는 Mobile IP 프로토콜을 설계하였다. 모바일 IP 프로토콜에는 기존의 라우터 및 단말의 수정 없이 이동성을 지원하는 베이스 모바일 IP(Base Mobile IP)^[1]와 베이스 모바일 IP를 사용하는 경우 발생하는 라우팅의 비효율성, 이른바 삼각 라우팅 문제(Triangle Routing Problem)^{[2][3]}를 해결하기 위한 경로 최적화 확장(Route Optimization Extension)이 있다^[4]. 베이스 모바일 IP에서는 이동단말의 홈 네트워크에 있는 HA(Home Agent)가 항상 이동단말(Mobile Host : MH)과 통신 중인 단말(Correspondent Host : CH)이 보내오는 패킷들을 가로채어 이동단말이 이동한 위치로 보낸다. 따라서, 기존의 라우터 및 단말의 수정 없이 이동성을 지원할 수 있다는 장점이 있으나, 패킷이 항상 HA를 경유하여 이동단말에 도착하기 때문에 경로가 길어지는 문제가 발생한다. 베이스 모바일 IP에서는 이동단말이 현재의 서브네트워크에서 다른 서브네트워크로 이동할 경우, HA는 이동단말의 이동한 위치를 이동단말이 보내오는 메시지를 통하여 알기 전까지는 패킷들을 이동단말이 있던 이전 위치로 보낸다. 따라서, 이동단말이 자신의 홈 네트워크로부터 멀리 떨어져 있는 경우에는 상당수의 패킷들이 핸드오프로 인하여 손실되는 문제가 발생한다. 이와 같이 사용자의 이동성으로 인하여 패킷 손실이 발생할 경우에는 모바일 IP의 상위 계층 프로토콜인 TCP는 심각한 영향을 받게 된다^[5]. 문제는 TCP가 원래 이동 네트워크 환경을 고려하지 않고 설계된 프로토콜이라는 점에 있다. 즉, 통신중인 사용자의 이동이 없는 유선 네트워크 환경에서는 패킷의 손실을 네트워크 혼잡의 신호로 해석하여도 무리가 없으나, 이동 네트워크 환경에서 패킷의 손실을 네트워크 혼잡의 신호로 해석하면 사용자의 이동성으로 인하여 패킷의 손실이 발생하는 경우에는 불필요하게 TCP 송신자의 패킷 전송 레이트를 줄이는 혼잡제어(Congestion Control)을 수행하게 됨으로써 TCP의 수율(Throughput)이 저하되는 문제가 발생한다. 특히, 이동단말의 사용자가 통신 중에 다수의 서브네트워크를 경유하게 될 경우에는 TCP의 성능이 심각하게 저하되는 문제가 발생한다.

한편, 베이스 모바일 IP가 내재하고 있는 라우팅의 비효율성 문제를 해결하기 위해 제안된 모바일 IP의 경로 최적화 확장에서는 핸드오프에 의한 패킷 손실 때문에 발생하는 TCP의 성능 저하 문제를 해결하기 위하여 스무스 핸드오프(Smooth Handoff)

메카니즘을 채택하고 있다. 스무스 핸드오프에서는 이동단말이 새로운 서브네트워크로 이동할 경우, 이동단말이 이동 후 위치한 서브네트워크에 있는 현재의 FA(Foreign Agent)를 통하여 이동 전에 머물던 서브네트워크에 있는 이전의 FA에게 알림으로써, 이동단말의 현재 위치를 모르는 통신 중인 단말(CH)이 보내오는 패킷들을 이전의 FA가 현재의 FA에게 포워딩할 수 있도록 한다. 따라서, 현재의 FA는 이전의 FA와 항상 인접해 있기 때문에 스무스 핸드오프를 사용하면 베이스 모바일 IP 경우와 비교하여 핸드오프에 의해 손실되는 패킷의 수는 감소하게 된다.

그러나 모바일 IP 표준의 경로 최적화 확장에 의한 스무스 핸드오프는 핸드오프 동안에 발생하는 패킷 손실을 줄이기 위하여 설계되었음에도 불구하고, 대부분의 경우 핸드오프 동안 발생하는 패킷 손실 패턴에 따른 TCP의 성능저하를 막지 못한다. 또한, RTT가 동일한 경우, 경로 최적화 확장의 스무스 핸드오프에서 오히려 패킷 손실이 많이 발생하는 베이스 모바일 IP를 사용한 경우보다 TCP의 성능저하가 심화된다^[6]. 이러한 경로 최적화 확장의 스무스 핸드오프에서 발생하는 TCP 성능 저하를 방지하기 위해서는 핸드오프 동안 4개 이상의 연속적인 패킷 손실을 방지하고 송신 중인 단말(CH)에게 중복된 TCP ACK 패킷이 3개 이상 연속적으로 수신되지 않도록 핸드오프 동안에 발생하는 패킷 손실을 복구하여 핸드오프가 전송 계층 프로토콜에 영향을 미치지 않게 하는 패킷 버퍼링 방식이 요구된다.

모바일 IP 망에서의 패킷 버퍼링 방식은 확장성(Scalability)을 지원하고 이미 이동단말이 수신한 패킷의 버퍼링을 방지하기 위해 ARQ(Automatic Repeat reQuest) 방식이 지원되는 기지국(Base Station : BS)에서 수행되어야 한다^{[7][8]}. 즉, 모바일 IP 프로토콜에서 핸드오프 동안에 발생하는 패킷 손실 때문에 초래되는 TCP의 성능 저하를 방지하기 위해서는 경로 최적화 확장의 스무스 핸드오프 메커니즘을 사용하고 이전 기지국에서 핸드오프 동안 손실된 패킷들을 버퍼에 저장하고 버퍼링된 패킷들이 이동단말에게 전달되어야 한다. 그러나 이동단말이 혼잡한 라우터와 연결된 새로운 서브네트워크로 이동한 경우에는 이전 기지국이 포워딩하는 패킷들은 손실되고, 포워딩된 패킷들로 가중된 혼잡도로 인하여 기존 고정 TCP 플로들의 링크 이용률 성능도 저하되게 된다. 현재 혼잡이 존재하는 이동네트워크

에서 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장이 미치는 영향을 분석한 기존 연구는 전무한 상태이다. 본 논문에서는 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식으로 이동단말이 혼잡이 존재하는 라우터와 연결된 새로운 서브네트워크로 이동한 경우, 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장이 미치는 영향을 분석하고 새로운 서브 네트워크와 연결된 라우터가 사용하는 RED 버퍼 관리 방식⁹⁾의 혼잡 상태에 따라 이전 기지국이 저장한 패킷들을 포워딩하거나 폐기하는 패킷 포워딩 제어방안을 제안하였다. 본 논문의 구성은 다음과 같다. 제 2 절에서는 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식에서 TCP 성능저하를 유발하는 핸드오프 시 패킷 손실 패턴 및 기지국 패킷 버퍼링 방식을 소개하고 이전 기지국에서 수행하는 패킷 포워딩 제어 방안을 제안한다. 제 3 절의 시뮬레이션 결과는 제안하는 패킷 포워딩 제어 방안을 적용하여 링크 이용률 성능을 향상시킬 수 있음을 보인다. 끝으로 제 4 절에서 결론을 맺는다.

II. 모바일 IP 프로토콜의 성능향상을 위한 패킷 포워딩 제어 방안

2.1. 모바일 IP 경로 최적화 확장에서 핸드 오프 시 TCP 성능저하를 유발하는 패킷 손실 패턴과 기지국 패킷 버퍼링 방식

본 논문에서는 그림 1과 같은 모바일 IP를 기반으로 하는 네트워크의 구성을 고려한다. 그림 1에서는 각 서브네트워크에 있는 라우터가 또한 FA

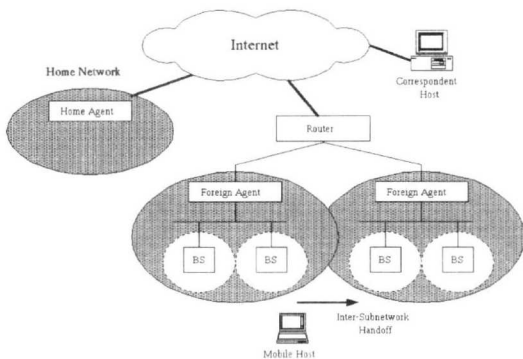


그림 1. 모바일 IP를 기반으로 하는 네트워크 구성

(Foreign Agent)의 역할을 수행한다고 가정한다. 그리고 모바일 IP 규격에서 사용되는 용어 및 기능 실체 (Functional Entity)에 관한 설명 없이 논의를 진행한다¹¹⁾¹³⁾. 그림 2의 베이스 모바일 IP 프로토콜에서는, 이동단말의 홈 네트워크에 있는 HA가 프록시 ARP (Proxy ARP)를 사용하여 이동단말을 목적지로 하는 패킷들을 가로챈다. 그 후에 HA는 이들 패킷들을 터널링 기법(Tunneling Technique)을 사용하여 이동단말의 인터넷에 대한 현재 접속지점 (Attachment Point)에 전달한다. 이동단말의 접속지점은 Care-of Address라 불리우는 IP 주소에 의하여 정의된다. 이동단말이 새로운 외부 네트워크 (Foreign Network)로 이동할 경우, 이동단말은 Care-of Address (즉 새로운 FA의 IP 주소)를 포함하고 있는 에이전트 광고 메시지(Agent Advertisement Message)를 새로운 FA로부터 수신한다. 이때, 이동단말은 수신된 에이전트 광고 메시지를 보고 자신이 새로운 외부 네트워크로 이동하였음을 알게 되고 이동단말은 새로운 FA에게 새로운 Care-of Address의 등록을 요청하는 등록 요청 메시지 (Registration Request Message)를 송신한다. 새로운 FA는 수신한 등록요청메시지를 이동단말의 HA에게 중계하여, HA가 이동단말을 목적지로 하는 패킷들을 이전 FA대신에 새로운 FA에게 전달할 수 있도록 한다.

그림 3의 모바일 IP의 경로 최적화 확장에서는, 새로운 FA가 이동단말로부터 등록 요청 메시지를 수신할 경우, 새로운 FA는 등록 요청 메시지를 HA로 중계하는 것과 더불어 이전 FA에게 바인딩 갱신 메시지 (Binding Update Message)를 보내어 새로운 Care-of Address를 통지한다¹⁴⁾. 따라서, 이전 FA는 아직 새로운 Care-of Address를 알지 못하는 대응단말 (Correspondent Host : CH) 즉, 이동단말과 통신 중인 단말로부터 패킷을 수신할 때마다, 수신된 패킷을 새로운 FA에게 전달한다. 또한, 이전

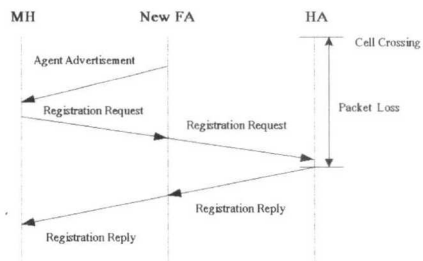


그림 2. 베이스 모바일 IP 프로토콜

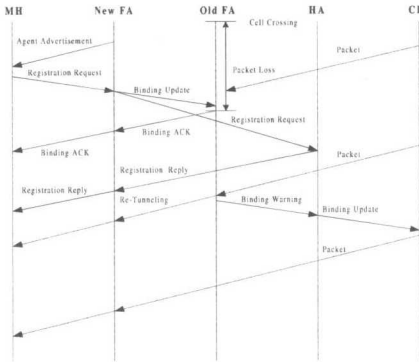


그림 3. 경로 최적화 확장을 갖는 모바일 IP 프로토콜

FA는 바인딩 경고 메시지 (Binding Warning Message)를 HA에게 보내어, HA가 대응단말에게 새로운 Care-of Address를 통지할 수 있도록 한다. HA가 바인딩 경고 메시지를 수신할 경우, HA는 바인딩 갱신 메시지를 대응단말에게 보내어 새로운 Care-of Address를 통지한다. 대응단말은 바인딩 갱신 메시지를 수신한 후에는 이전 FA 대신에 새로운 FA에게 패킷을 보낼 수 있게 된다. 일반적으로 새로운 FA와 이전 FA간의 거리가 새로운 FA와 HA간의 거리에 비해 훨씬 짧기 때문에, 새로운 Care-of Address를 이전 FA에게 통지함으로써 핸드오프 동안에 손실되는 패킷의 수를 줄일 수 있다. 모바일 IP에서는 이를 스무스 핸드오프라 부른다.

모바일 IP 표준은 에이전트 광고 메시지에 의해 발생하는 네트워크의 부하를 줄이기 위하여 에이전트 광고 메시지의 최대 송신 레이트를 초당 1 회로 제한할 것을 권고한다^[1]. 따라서 에이전트 광고 메시지가 새로운 FA에 의하여 최대한의 송신 레이트

로 방송 혹은 멀티캐스팅 된다 할지라도, 이동단말은 새로운 외부 네트워크에 이동한 후에 최악의 경우 1 초 동안 에이전트 광고 메시지를 수신할 수 없다. 이동단말은 에이전트 광고 메시지를 수신하기 전에는 등록 요청 메시지를 송신할 수 없기 때문에, 이는 최악의 경우 1 초 이상 이동단말을 향하여 전달되고 있는 패킷들이 손실될 수 있음을 의미한다. 이러한 문제 때문에 참고문헌 [6]에서는 그림 4의 프로토콜을 제안하였고, 그림 4에서는 동일 서브네트워크 안에 존재하는 기지국 (BS: Base Station)간의 핸드오프를 수행하는 로컬 핸드오프 프로토콜 (Local Handoff Protocol)을 모바일 IP의 경로 최적화 확장에 결합하였다. 이동단말은 에이전트 광고 메시지와 비슷한 역할을 하는 비콘 메시지(Beacon Message)를 새로운 기지국으로부터 수신한 후에 핸드오프 요청 메시지(Handoff Request Message)를 새로운 기지국에 보낸다. 새로운 기지국은 그후에 통보 메시지(Notification Message)를 새로운 FA에게 보내어 에이전트 광고 메시지를 요청한다. 새로운 FA는 통보 메시지를 받자마자 새로운 기지국에게 에이전트 광고 메시지를 보낸다. 이러한 방법은 서브네트워크에 존재하는 모든 기지국에게 주기적으로 에이전트 광고 메시지를 방송하는 방법에 비하여 훨씬 빠르게, 이동단말이 에이전트 광고 메시지를 수신할 수 있도록 하여 패킷 손실 시간을 비콘 메시지 주기로 감소시킬 수 있다. 왜냐하면, 일반적으로 로컬 핸드오프 프로토콜에서 사용되는 비콘 메시지는 에이전트 광고 메시지에 비하여 훨씬 짧고 비콘 메시지의 송신 레이트 또한 에이전트 광고 메시지의 송신 레이트에 비하여 훨씬 높기 때문이다. 모바일 IP 표준은 이러한 로컬 핸드오프 프로토콜과의 협력을 허용한다는 것을 주목하기 바란다^[1]. 또한 그림 4의 프로토콜은 기지국 패킷 버퍼링 방식을 지원한다. 이를 위해 이전 FA가 바인딩 갱신 메시지를 수신하는 시점에서 이전 기지국이 핸드오프 동안 버퍼링한 패킷들을 새로운 FA로 포워딩하도록 지시하는 Forward 메시지가 추가되어 모바일 IP 프로토콜이 수정되었다^{[6][10]}.

로컬 핸드오프 프로토콜이 결합된 베이스 모바일 IP 프로토콜과 모바일 IP 경로 최적화 확장에 대한 패킷 손실 시간 T_{loss} 및 연결 왕복시간(Round Trip Time) T_{RTT} 와 전체 핸드오프 소요시간 T_h 에 대해 정리하면 식(1)과 같다^[6]. 전체 핸드오프 소요시간은 베이스 모바일 IP 프로토콜의 경우 이동단말의 이동 시점에서 HA가 새로운 FA로부터 등록 요청 메

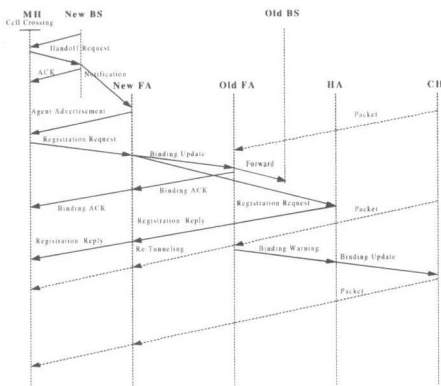


그림 4. 기지국 패킷 버퍼링 방식을 지원하는 모바일 IP 프로토콜의 경로 최적화 확장

시지를 수신하는 시점까지이고, 경로 최적화 확장의 경우에는 이동단말의 이동시점에서 대응단말(CH)이 HA로부터 바인딩 갱신 메시지를 받는 시점까지이다. 즉, 전체 핸드오프 소요시간 동안 베이스 모바일 IP 프로토콜에서는 패킷 손실이 발생하게 되고, 경로 최적화 확장의 경우에는 전체 핸드오프 소요시간 동안 대응단말이 송신한 패킷들이 이전 FA를 경유하여 새로운 FA로 전달되게 된다. 한편 이동단말은 TCP의 ACK 패킷들을 대응단말에게 직접 송신한다. 식(1)로부터 T_{loss} 는 경로 최적화 확장의 경우보다 베이스 모바일 IP의 경우에서 훨씬 커짐을 알 수 있다. 왜냐하면, 이전 FA는 항상 새로운 FA에 인접해 있음으로 T_O 의 값이 매우 작아지기 때문이다. 따라서, 두 경우에 있어서 T_{loss} 의 차는 대략 HA와 이전 혹은 새로운 FA 간 지연 시간의 두 배와 같다.

$$T_{loss-Base} = T_B + T_N + T_{HA} + D_{HA-FA}$$

$$T_{loss-Route} = T_B + T_N + T_O$$

$$T_{h-Base} = T_B + T_N + T_{HA}$$

$$T_{h-Route} = T_B + T_N + T_{HA} + T_{CH}$$

$$T_{RTT-Base} = D_{CH-HA} + D_{HA-FA} + D_{CH-FA} + \alpha$$

$$T_{RTT-Route} = 2 \cdot D_{CH-FA} + \alpha$$

D_{HA-FA} : HA와 이전 FA 간의 지연 ($\cong T_{HA}$: HA와 새로운 FA 간의 지연)

D_{CH-HA} : CH와 HA간의 지연, D_{CH-FA} : CH와 FA간의 지연

T_B : 이동단말이 새로운 외부 네트워크에 진입한 시점부터 새로운 BS로부터 비이콘 메시지를 수신하는 시점까지의 시간

T_N : 이동단말이 비이콘 메시지를 수신한 시점부터 새로운 FA가 등록 요청 메시지를 송신하는 시점까지의 시간 (셀이 겹치지 않는 경우, 최대 비이콘 메시지의 주기)

T_{HA} : 새로운 FA가 등록 요청 메시지를 송신하는 시점부터 HA가 이를 수신하는 시점까지의 시간

T_O : 새로운 FA가 바인딩 메시지를 송신하는 시점부터 이점 FA가 이를 수신하는 시점까지의 시간

T_{CH} : HA가 바인딩 갱신 메시지를 송신하는 시점부터 CH가 이를 수신하는 시점까지의 시간

α : FA와 이동단말 간의 왕복시간 (1)

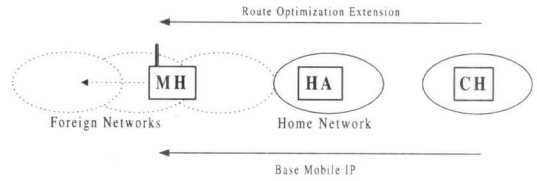


그림 5. 스무스 핸드오프 효과를 알아보기 위한 시나리오

위에서 소개한 베이스 모바일 IP 방식과 모바일 IP 경로 최적화 확장 방식에 대해 참고문헌[6]은 모바일 IP의 경로 최적화 확장에 의한 스무스 핸드오프가 TCP의 성능에 미치는 영향을 알아보기 위하여, 이동단말이 HA와 대응단말로부터 점점 멀어지는 그림 5와 같은 시나리오를 고려하였다. 상기 시나리오에서 이동단말과 대응단말 간에 설정된 TCP 연결의 왕복시간은 두 모바일 IP 프로토콜에 대하여 동일하다. 즉, 상기 시나리오에서는 경로 최적화가 TCP의 성능에 미치는 영향을 무시할 수 있으며, 따라서 스무스 핸드오프가 TCP의 성능에 미치는 영향에만 집중할 수 있다. 그러나 상기 시나리오에서 모바일 IP 표준의 경로 최적화 확장에 의한 스무스 핸드오프는 핸드오프 동안에 발생하는 패킷 손실을 줄이기 위하여 설계되었음에도 불구하고 베이스 모바일 IP보다 낮은 TCP 수율을 나타내었다 [6].

이와 같은 결과의 원인을 알아보기 위하여 핸드오프가 발생할 때 TCP Tahoe가 어떤 식으로 행동하는지를 그림 6의 예에서 살펴본다. 패킷 5 ~ 8 들이 핸드오프 동안에 하나의 윈도우에 상응하는 데이터로부터 손실되며 나머지 패킷 9 ~ 12 들은 TCP 수신자가 성공적으로 수신한다고 가정한다. 이와 같은 경우에는 TCP 송신자는 패킷 4에 대한 중복된 ACK를 연속적으로 3 번 수신하자마자 패킷 5를 재전송한다. 왕복시간이 두 번 지난 후에 (즉, 기간 T_1) TCP 송신자는 패킷 8 ~ 11 들에 의하여 촉발된 패킷 12에 대한 4 개의 ACK들을 수신한다. 패킷 12에 대한 첫 번째 ACK를 수신하였

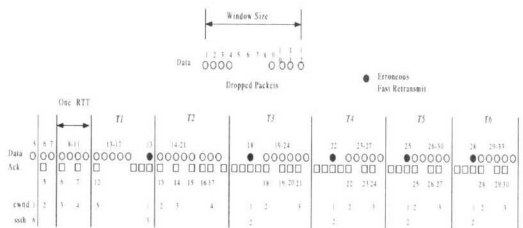


그림 6. 다수의 패킷손실로 인한 잘못된 Fast Retransmit 예

을 때, 이시점에서 혼잡 윈도우의 크기가 5이기 때문에 TCP 송신자는 응답으로써 패킷 13 ~ 17 들을 송신한다. 이후에 TCP 송신자는 패킷 12에 대한 중복된 ACK를 연속적으로 3 번 수신하자마자, 바로 전에 패킷 13 ~ 17 들이 이미 송신되었음에도 불구하고 패킷 13을 복구하기 위하여 Slow-start를 시작한다. 이에 대한 결과로서 하나의 왕복시간이 지난 후에, TCP 송신자는 하나의 ACK 대신에 다섯 개의 ACK를 수신하게 되며, 따라서 단지 두 개의 패킷이 전송되는 정상적인 Fast Retransmit와는 달리 8 개의 패킷들 (패킷 14 ~ 21)을 송신한다. TCP 송신자는 Slow-start를 할 때 패킷 14 ~ 17 들이 이전에 송신되었다는 사실을 무시하기 때문에 이들 패킷들은 다시 전송됨에 주목하기 바란다. 상기와 같은 이유로 하나의 왕복시간에 해당하는 시간이 지난 후에 잘못된 Fast Retransmit가 발생한다. 발생한 잘못된 Fast Retransmit는 다시 하나의 왕복시간에 해당하는 시간이 지난 후에 또 다른 잘못된 Fast Retransmit를 초래한다. 이와 같은 방식으로 매 왕복시간 마다 잘못된 Fast Retransmit가 파일 전송이 완료될 때까지 반복적으로 발생한다. 첫 번째 잘못된 Fast Retransmit가 발생한 후에, 하나의 왕복시간 동안에 전송되는 패킷의 수는 6이 될 때까지 점점 줄어들게 되며 이후에는 변하지 않는다. 또한, 잘못된 Fast Retransmit가 발생할 때에 하나의 왕복시간 동안에 혼잡 윈도우의 크기보다 많은 패킷들이 전송됨을 알 수 있다. 이로 인하여 모든 패킷이 두 번씩 전송된다. 이와 같은 잘못된 Fast Retransmit는 다음과 같은 식(2)의 두 가지 조건이 동시에 만족되면 연속적으로 발생한다⁶⁾.

1. 핸드오프 동안에 4 개 이상의 패킷들이 연속적으로 손실된다.
 2. 4개 이상의 연속된 패킷 손실 후 3개이상의 패킷들이 성공적으로 TCP 수신자에게 도착한다.
- (2)

불행하게도 이와 같은 조건들은 모바일 IP의 경로 최적화 확장이 사용되는 경우에는 만족된다. 이로 인하여 경로 최적화 확장이 사용될 때에는 베이스 모바일 IP를 사용하는 경우보다 TCP의 성능이 더욱 악화된다. 한편, 베이스 모바일 IP의 경우에는 왕복시간에 관계없이 핸드오프 동안 윈도우 크기에 해당하는 패킷들이 항상 손실되고, 경로 최적화 확장을 사용할 때에는 왕복시간 $T_{RTT-Route}$ 이 $T_{loss-Route}$ 보다 커질 경우, 잘못된 Fast Retransmit를

연속적으로 초래하는 조건들이 만족된다⁶⁾. 그러나, 베이스 모바일 IP의 경우에는 윈도우의 크기에 해당하는 패킷들이 모두 손실되기 때문에 TCP 수신자는 중복된 ACK를 보내지 않는다. 따라서, TCP 송신자는 재전송 타임아웃 후에 파일 전송을 다시 시작하며 잘못된 Fast Retransmit는 일어나지 않는다. 이상과 같은 이유로 인해, 스무스 핸드오프의 효과를 알아보기 위한 시나리오에서 베이스 모바일 IP는 경로 최적화 확장보다 우수한 성능을 나타낸다. 경로 최적화 확장에 의한 스무스 핸드오프는 TCP의 왕복시간 $T_{RTT-Route}$ 이 $T_{loss-Route}$ 보다 훨씬 커서 4 개 이상의 패킷이 손실되지 않을 경우에만 효과가 있다. 이 경우에는 손실된 모든 패킷들이 정상적인 Fast Retransmit에 의하여 복구되고 잘못된 Fast Retransmit는 일어나지 않기 때문에, 경로 최적화 확장은 베이스 모바일 IP 보다 우수한 성능을 나타낸다.

참고문헌[6]에서는 모바일 IP 프로토콜의 경로 최적화 확장을 사용하는 이동 네트워크에서 핸드오프로 인한 TCP의 성능저하 문제를 근본적으로 해결하기 위해 그림 4와 같이 핸드오프 동안에 손실되는 패킷들을 기지국에서 저장하여 이를 이동단말에게 포워딩하는 패킷 버퍼링 방식을 제안하였다. 기지국 패킷 버퍼링 방식은 다수의 TCP 플로우를 관리해야 하는 FA 패킷 버퍼링 방식과 달리 확장성의 문제없이 모바일 IP 네트워크에 적용될 수 있다. 또한, 기지국 패킷 버퍼링 방식은 ARQ 기법을 사용하여 이미 이동단말이 수신한 패킷의 버퍼링을 방지할 수 있다. 참고문헌[10]에서는 경로 최적화 확장 방식에서 아직 새로운 Care-of Address를 알지 못하는 대응단말로부터 이전 FA가 수신한 패킷들과 이전 기지국에서 버퍼링한 패킷들이 새로운 FA로 포워딩되는 경우를 고려하였다. 그리고 이 경우 이동단말이 수신할 때 발생할 수 있는 패킷 순서의 재배열 문제(Packet Reordering)는 식(2)의

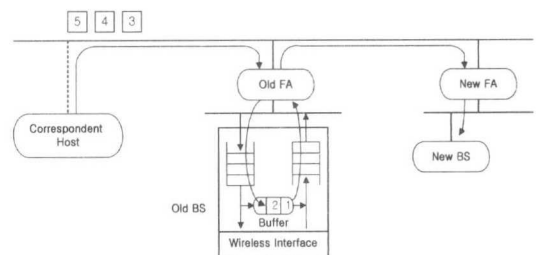


그림 7. 패킷재배열 문제를 해결하기 위한 패킷포워딩 순서

TCP 성능저하 패킷 손실 패턴과 동일한 현상을 야기할 수 있고 이를 해결하기 위해 그림 7과 같이 아직 새로운 Care-of Address를 알지 못하는 대응 단말로부터 이전 FA가 수신한 패킷들은 반드시 이전 기지국을 경유하여 새로운 FA로 포워딩되도록 제안하였다. 본 논문에서는 기지국 버퍼링 방식을 사용하고 ARQ 방식을 이용하여 이미 이동단말이 수신한 패킷을 버퍼링하는 것은 방지되었다고 가정하였으며, 이전 FA가 수신한 패킷들은 반드시 이전 기지국을 경유하여 새로운 FA로 포워딩 되도록 설정하였다.

2.2. 모바일 네트워크에서 TCP 성능을 향상시키기 위한 패킷 포워딩 제어 방안

2.1절에서는 모바일 IP 경로 최적화 확장에서 핸드오프 시 TCP 성능저하를 유발하는 패킷 손실 패턴과 TCP 성능저하를 방지하기 위한 기지국 패킷 버퍼링 방식을 소개하였다. 그러나 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식으로 이동단말이 혼잡한 라우터와 연결된 새로운 서브네트워크로 이동한 경우에는 이전 기지국이 포워딩하는 패킷들은 손실되고, 포워딩된 패킷들로 가중된 혼잡도로 인하여 기존 고정 TCP 플로들의 링크 이용률 성능도 저하되게 된다. 본 절에서는 이동단말이 혼잡이 존재하는 라우터와 연결된 새로운 서브네트워크로 이동한 경우, 새로운 서브 네트워크와 연결된 라우터가 사용하는 RED 버퍼 관리 방식의 혼잡 상태에 따라 이전 기지국이 저장한 패킷들을 포워딩하거나 폐기하는 패킷 포워딩 제어 방안을 제안하였다.

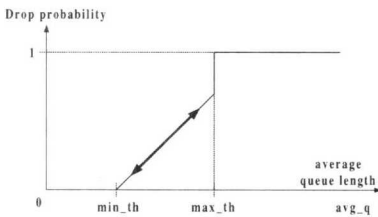


그림 8. RED 버퍼 관리 방식

현재 네트워크에서는 일반적으로 혼잡제어를 위해 그림 8의 RED 버퍼관리방식을 사용하고 있다⁹⁾. RED 알고리즘을 구성하는 제어 변수로는 min_th , max_th 와 avg_q , P_{max} , W_q 가 있고 참고문헌⁹⁾에서는 P_{max} 와 W_q 의 값으로 각각 0.02와 0.002를 제시하였다. RED 버퍼관리방식은 min_th 와 max_th 라는

2개의 임계치를 설정하여 평균 큐의 길이를 나타내는 $avg-q$ 변수 값이 min_th 를 초과하게 되면 확률적으로 도착하는 패킷들을 폐기하고 max_th 에 이르면 무조건 도착하는 패킷들을 폐기한다. 이러한 RED 버퍼관리 방식은 버퍼가 가득 차기 전에 큐에 도착하는 패킷들을 확률적으로 폐기하여 평균 큐 길이를 제어할 수 있고 따라서 패킷들의 평균 지연 시간을 제어할 수 있다. 따라서 RED 방식의 경우, Tail-Drop 방식에서 많은 수의 플로들에서 패킷들이 동시에 폐기되어 네트워크의 링크 대역폭에 대한 이용효율을 감소시키는 Global Synchronization의 문제점을 보완하여 수율 및 링크 이용률을 향상시킬 수 있다.

그림 9는 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식으로 이동단말이 혼잡한 라우터와 연결된 새로운 서브네트워크로 이동한 경우를 나타낸 것이다. 현재 기지국 버퍼링 방식에서는 그림 9와 같이 이전 FA(old FA)가 새로운 FA(New FA)로부터 바인딩 갱신 메시지를 수신하면 아무런 제어 없이 이전 기지국(old BS)은 버퍼링된 패킷들을 혼잡한 라우터 CR을 경유하여 새로운 FA로 포워딩하게 된다. 제안하는 패킷 포워딩 제어 방안에서는 이전 FA가 새로운 FA로부터 라우터 CR의 최근 RED 상태 정보를 포함한 바인딩 갱신 메시지를 수신하여 이전 기지국에게 CR의 RED 상태 정보를 포함한 그림 4의 Forward 메시지를 전달하면, 이전 기지국은 수신한 CR의 RED 상태 정보를 바탕으로 버퍼링된

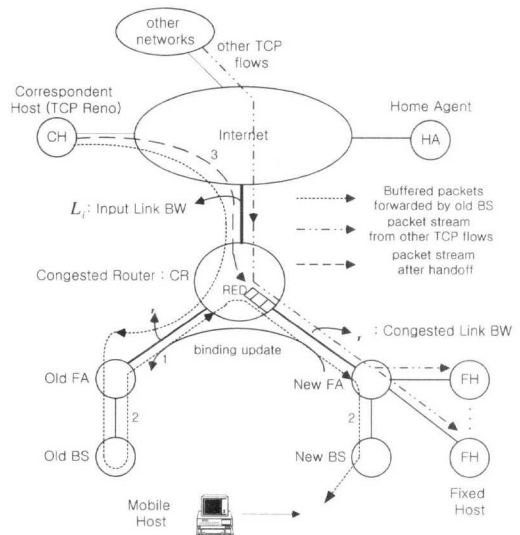


그림 9. 혼잡한 라우터를 경유하는 버퍼링된 패킷의 포워딩

패킷들의 폐기 또는 포워딩을 결정한다. 즉, 제안하는 방식에서는 새로운 FA로부터 발생한 바인딩 갱신 메시지가 혼잡한 CR 라우터를 경유할 때 새로운 FA와 연결된 링크를 관리하는 RED의 상태 정보를 바인딩 갱신 메시지에 담아 이전 FA에게 전달할 수 있고 이전 FA는 전달 받은 RED 상태 정보를 Forward 메시지에 담아 이전 기지국에게 전달할 수 있도록 모바일 IP 프로토콜을 수정하였다. 제안하는 방식에서 바인딩 갱신 메시지가 포함하게 되는 CR 라우터의 현재 t_{bind} 시점에서의 RED 큐 상태 정보는 $avg-q$ 변수 값 $avg-q-bind$ 와 큐의 길이 $q-bind$ 이다. 그러나 $avg-q-bind$ 와 $q-bind$ 의 정보는 이전 기지국이 버퍼링한 패킷들을 포워딩하여 포워딩된 패킷들이 CR 라우터를 경유할 때의 RED 상태 정보를 정확하게 반영하지 않는다. 그림 9에서 CR 라우터와 이전 기지국(Old BS)간의 링크 지연시간을 D_o 라 하면 포워딩된 패킷들이 CR 라우터를 경유하는 시작 시점은 $t_{bind} + 2 \cdot D_o$ 가 된다.

한편 핸드오프동안 이전 기지국에서 버퍼링되는 최대 패킷 수는 최대 윈도우 크기(Maximum Window Size), mus 에 해당하므로 포워딩된 패킷들이 CR 라우터를 경유하는 최종 시점은 $2 \cdot D_o + mus \cdot packetsize / L_o$ 가 된다. 따라서 제안하는 포워딩 제어 방안에서는 바인딩 갱신 메시지가 전달하는 t_{bind} 시점에서 $avg-q-bind$ 와 $q-bind$ 의 정보를 기반으로 $t_{bind} + 2 \cdot D_o + mus \cdot packetsize / L_o$ 시점에서의 CR 라우터의 혼잡 상태를 추정하여 이전 기지국이 버퍼링한 패킷들의 폐기 또는 포워딩을 결정한다.

이전 기지국에서 포워딩하는 패킷들이 혼잡한 CR 라우터에서 폐기되는 경우에는 그림 9에서 CR 라우터와 새로운 FA간의 L_n 링크를 관리하는 RED 큐가 오버플로우(Overflow) 되는 경우와 CR 라우터의 RED 변수 $avg-q$ 값이 $max\ th$ 이상일 경우가 있다. 식(3)은 바인딩 갱신 메시지가 전달하는 t_{bind} 시점에서 $avg-q-bind$ 와 $q-bind$ 의 정보를 기반으로 $t_{bind} + 2 \cdot D_o + mus \cdot packetsize / L_o$ 시점에서 L_n 링크를 관리하는 RED 큐의 오버플로우 여부를 추정하는 식이다. 식(3)은 그림9에서 CR 라우터와 연결된 입력 링크 대역폭의 총합 L_i 의 최대 데이터 도착률이 $2 \cdot D_o + mus \cdot packetsize / L_o$ 의 시간 동안 β 의 확률로 지속된다고 가정하였다. 그리고 L_i 에서 혼잡한 링크 대역폭의 크기 L_n 을 뺀 $L_i - L_n$ 의 최대 속

도로 $2 \cdot D_o + mus \cdot packetsize / L_o$ 의 시간동안 RED 큐에 패킷이 버퍼링된 양과 mus 의 패킷들이 버퍼링된 양을 t_{bind} 시점에서의 $q-bind$ 값과 합하여 L_n 링크를 관리하는 RED 큐의 버퍼크기, $buff-size$ 와 크기를 비교하여 오버플로우 되는 패킷양, n_{overs} 값을 구하는 식이다. 제안하는 패킷 포워딩 제어 방안에서는 식(3)에서 추정된 오버플로우되는 패킷 양 n_{overs} 이 mus 보다 작을 경우 mus 에서 $n_{overs}(\geq 0)$ 를 뺀 $mus - n_{overs}$ 만큼만 이전 기지국에서 버퍼링한 패킷들에서 시퀀스 번호가 가장 앞인 패킷들부터 포워딩한다. 따라서 혼잡한 라우터에서 이전 기지국이 포워딩하는 패킷들로 인해 기존 TCP 플로우들이 겪는 TCP 수율 성능 저하를 완화할 수 있다.

$$n_{over} = \left\{ q_bind + \beta \cdot \left(\frac{(2 \cdot D_o + mus \cdot packet_size / L_o)(L_i - L_n)}{packet_size} \right) + mus \right\} - buff_size$$

$$arr_{estimated} = \beta \cdot \left(\frac{(2 \cdot D_o + mus \cdot packet_size / L_o) \cdot L_i}{packet_size} \right) + mus - n_{over}$$

if $0 \leq n_{over} \leq mus$

$arr_{estimated}$: $2 \cdot D_o + mus \cdot packetsize / L_o$ 시간동안 L_n 링크에 도착하는 패킷량의 추정값 (3)

식(4)는 혼잡한 CR 라우터의 RED 변수 $avg-q$ 가 $max\ th$ 이상의 값을 가짐으로써 이전 기지국에서 포워딩하는 패킷들이 폐기되는 것을 추정할 알고리즘이다. 식(4)는 RED 알고리즘으로부터 t_{bind} 시점에서 $avg-q$ 가 $avg-q-bind$ 의 값을 가질 때 t_{bind} 시점에서 $2 \cdot D_o + mus \cdot packetsize / L_o$ 의 시간 동안 식(3)에서 나타낸 추정값 $arr_{estimated}$ 의 패킷양이 도착할 경우 $t_{bind} + 2 \cdot D_o + mus \cdot packetsize / L_o$ 시점에서 $avg-q$ 가 갖는 최대값을 추정하는 알고리즘을 나타낸다. 식(4)에서는 L_i / L_n 의 값을 λ_{Toq} 로 나타내었고, $arr_{estimated}$ 의 패킷들이 β 의 확률을 갖고 모두 $2 \cdot D_o + mus \cdot packetsize / L_o$ 시간의 앞부분에서 버스트하게 도착하여 $avg-q$ 가 최대값을 갖게 되는 경우를 나타낸다. 그리고 $avg-q$ 가 $max\ th$ 를 초과하지 않는 허용가능 패킷 도착량 $arr_{allowed}$ 를 도출하여 $avg-q$ 가 $max\ th$ 를 초과하게 되는 도착량 $n_{max\ th}(\leq mus)$ 을 구한다. 제안하는 기지국 패킷 포워딩 제어 방안에서는 식(4)의 알고리즘으로부터 구한 $n_{max\ th}$ 만큼의 패킷 수를 또한 폐기한다. 결과적으로 제안하는 패킷 포워딩 방안에서 이전 기지국이 포워딩하게 되는 패킷의 최대 수는 $mus - n_{overs} - n_{max\ th}$

가 된다. 식(3)과 식(4)를 포함한 제안하는 기지국 패킷 포워딩 제어 방안은 식(3)의 오버플로우에 의한 폐기에 우선 순위를 두어 전체적으로 식(5)와 같다.

```

arr_allowed = 0
q_length = q_bind
max_avg_q = avg_q_bind

if max_avg_q ≥ max_th
then arr_allowed = 0
else
while i = 1 to arr_estimated
when j · λ_top packets arrive
q_length = q_length - 1
when a packet arrives
max_avg_q = max_avg_q · (1 - w_q) + w_q · q_length
q_length = q_length + 1

if β · max_avg_q ≥ max_th
then exit
else arr_allowed = arr_allowed + 1
end
end

n_max_th = arr_estimated - arr_allowed
    
```

```

// packet forwarding control scheme at old Base station //

q_bind and avg_q_bind are given by binding update message
n_over and arr_estimated are calculated
arr_allowed and n_max_th are calculated
(mws - n_over - n_max_th) packets are forwarded by Old Base Station
    
```

(5)

III. 시뮬레이션 모델 및 성능 평가

본 논문에서는 혼잡이 존재하는 이동네트워크에서 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장이 미치는 영향을 분석하고, 제안하는 패킷 포워딩 제어 방안의 성능을 분석하기 위하여 그림 10의 시뮬레이션 모델을 제시하였다. 그리고 시뮬레이션 모델에서 이동단말의 핸드오프 소요시간 및 RTT를 구성하는 각 노드 간의 지연시간은 표1에 나타내었다. 표1에서 3Mbps의 전송대역 폭을 갖는 이전 기지국(Old BS)과 이동단말(MH) 간의 무선 링크 상에서 비이콘 메시지의 주기, T_B 는 50 msec로 고정하였다^[7]. 시뮬레이션 모델에서 각

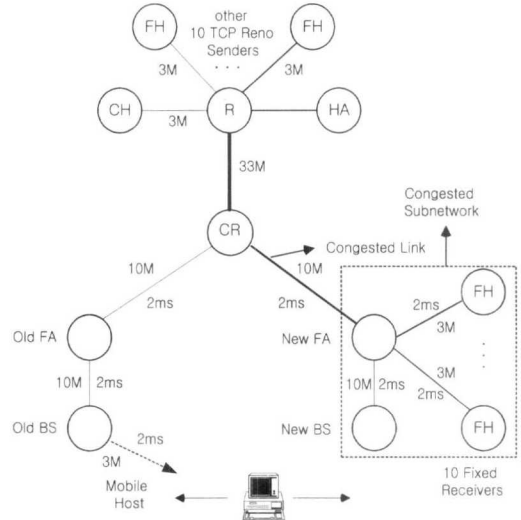


그림 10. 패킷 포워딩 제어방안에 대한 시뮬레이션 모델

고정 단말 간의 RTT는 2ms씩 차이가 나며 62ms에서 80ms까지의 값을 갖는다. 시뮬레이션에서는 모든 이동단말 및 고정 단말의 TCP 연결에 있어서 TCP Reno에서 채택된 혼잡 제어 알고리즘을 고려하였고 TCP 연결의 패킷 크기, packet-size 및 최대 윈도우 크기, mus 는 각각 512 바이트 및 64 패킷으로 설정하였다. 그리고 전송 에러로 인한 패킷 손실은 없다고 가정하였고 CR 라우터와 연결된 혼잡한 링크 L_n 을 관리하는 RED 큐의 버퍼 길이, $buff-size$ 는 300 패킷으로 min_{th} 와 max_{th} 는 각각 130, 260으로 P_{max} 와 W_q 의 값은 0.02와 0.002로 설정하였다^[9]. 본 논문에서는 ns-2 시뮬레이터를 사용하여 시뮬레이션을 수행하였다^[11].

표 1. 핸드오프 지연시간을 구성하는 각 노드간 지연시간

변수	T_B	T_N	T_{HA}	T_{CH}	T_o
값	50ms	12ms	60ms	60ms	4ms
변수	D_{CH-FA}	α	$T_{h-Route}$	$T_{loss-Route}$	$T_{RTT-Route}$
값	30ms	8ms	182ms	66ms	68ms

그림 11은 그림 10의 기지국 버퍼링 방식을 결합한 경로 최적화 확장 방식에서 이전 기지국이 포워딩하는 패킷들의 버스트한 도착 특성이 혼잡이 존재하는 이동네트워크에 미치는 영향을 CR 라우터에서의 RED 큐 길이의 변화 측면에서 분석한 결과이다. 즉, 기지국 버퍼링 방식을 결합한 경로 최적화 확장 방식에서 이전 기지국이 포워딩하는 패킷양의

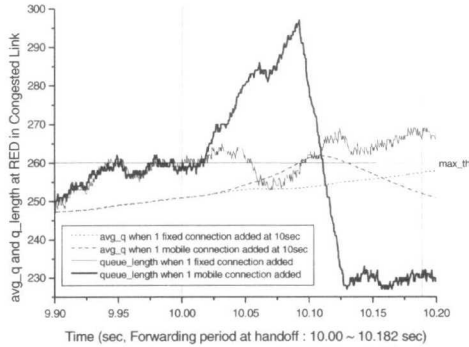


그림 11. 이전 기지국이 포워딩하는 패킷들의 버스트한 도착 특성

최대값은 mus 로서 64패킷이고 10Mbps의 L_0 링크에서 약 26ms의 짧은 시간 동안 혼잡한 CR 라우터에 버스트하게 포워딩된다. 이러한 특성으로 인해 그림 11에서와 같이 CR 라우터를 경유하는 기존 10개의 고정단말 플로들에게 새로운 고정 단말 플로 하나가 추가되는 경우보다 새로운 이동단말 플로 하나가 추가되는 경우가 RED 큐 길이 및 $avg-q$ 를 더욱 급격하게 증가시켜 RED 큐의 혼잡을 증가시키고 혼잡한 이동네트워크를 경유하는 플로들의 TCP 성능을 저하시키게 된다. 즉, 그림 11은 추가된 새로운 고정단말 플로로부터 발생된 패킷들이 10초에서부터 CR라우터의 혼잡된 링크에 도착하는 경우와 이동단말이 이동하여 이전 기지국이 버퍼링한 최대 mus , 64개 패킷들이 10초에서부터 182ms의 핸드오프 소요시간, $T_{h-Route}$ 동안 10.182초까지 CR라우터의 혼잡된 링크에 도착한 경우를 혼잡한 링크를 관리하는 RED 큐의 큐 길이 및 $avg-q$ 측면에서 분석한 결과이다. 그림 11의 결과에서 CR 라우터를 경유하는 기존 10개의 고정단말 플로들에게 새로운 고정 단말 플로 하나가 추가되는 경우에는 큐 길이와 $avg-q$ 가 점진적으로 증가하는 것을 볼 수 있으나, 새로운 이동단말 플로 하나가 추가되는 경우에는 큐 길이 및 $avg-q$ 가 급격하게 증가하여 10초에서부터 10.182초까지의 패킷 포워딩 기간 동안 $avg-q$ 가 $max.th$ 를 초과하게 되고 도착하는 패킷들에 대한 버스트한 폐기로 인해 큐 길이가 급격하게 감소하는 Tail-Drop의 Global Synchronization의 현상이 발생함을 알 수 있다.

표 2는 그림 10의 혼잡한 링크를 10개 플로들이 통과하고 있는 상황에서 기지국 버퍼링 방식이 지

표 2. β 값에 따른 포워딩 제어 방안의 링크이용률 성능

β	0.2 - 0.4	0.5	0.6	0.7
100초간 통과된 패킷 수	225810	223015	225323	225956
100초간 손실된 패킷 수	637	652	615	623
β	0.8	0.9	1.0	-
100초간 통과된 패킷 수	226958	228079	224561	-
100초간 손실된 패킷 수	575	695	630	-

원되는 이동단말이 100초간 두개의 서브네트워크로 총 10회 이동하여 혼잡한 새로운 서브네트워크로 이동단말이 5회 이동할 경우, 제안하는 포워딩 제어 방안의 성능을 설정한 β 값에 따라 혼잡한 링크 L_n 에서 100초간 통과된 패킷 수와 폐기된 패킷 수 측면에서 분석한 결과이다. 표 2에서와 같이 식(3)에서 L_n 의 최대 데이터 도착률의 발생 확률, β 값을 0.8로 하여 혼잡 상태를 추정하여 이전 기지국에서 버퍼링된 패킷들을 폐기한 경우에 손실된 패킷 수가 가장 작고, 링크 이용률도 β 값을 0.9로 하여 혼잡 상태를 추정한 경우 다음으로 큰 값을 나타내었다. 표 2의 결과로부터 그림 12에서는 100초간 혼잡된 서브네트워크로 이동한 회수에 따라 새로운 FA로 버퍼링된 패킷들을 무조건 포워딩하는 방식과 제안하는 포워딩 제어 방안의 성능을 β 값이 0.8인 경우와 0.9인 경우에서 혼잡한 링크 L_n 을 100초간 통과한 패킷 수 측면에서 분석한 결과이다. 그림 12

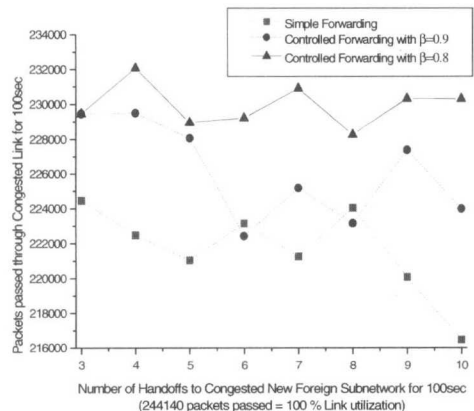


그림 12. 100초간 혼잡한 서브 네트워크로 이동한 회수에 따른 포워딩 제어 방안의 링크 이용률 성능

의 결과를 보면 β 값이 0.9인 경우에는 과도한 혼잡 상태 추정의로 오류로 무조건 포워딩하는 방식보다 링크 이용률이 작은 경우가 발생하나, β 값이 0.8인 경우에는 0.9인 경우와 무조건 포워딩하는 방식보다 링크 이용률이 높음을 알 수 있다. 이러한 현상은 β 값이 0.8인 경우가 β 값이 0.9인 경우보다 혼잡 상태를 덜 심화된 상태로 추정하여 이동단말의 버퍼링된 패킷을 더 적게 폐기함으로써 이동단말의 링크 이용률이 더 높기 때문이다. 그리고 이동단말 하나를 고려한 그림 12의 결과로부터 이동단말의 수가 더 많은 환경에서 제안하는 포워딩 제어 방안을 적용한다면 링크 이용률 성능은 더욱 크게 향상될 것이다.

한편 그림 13은 그림 12의 환경에서 새로운 FA로 버퍼링된 패킷들을 무조건 포워딩하는 방식과 β 값이 0.8인 제안하는 포워딩 제어 방안의 성능을 이동단말의 100초간 평균 수율 측면에서 분석한 결과이다. 단순히 무조건 포워딩하는 경우에는 그림 11에서와 같이 기존 연결 플로들에게 심각한 경우 Global Synchronization의 영향을 발생시킬 수 있다. 따라서 이동단말 플로가 기존 플로들과 동등하게 데이터를 전송할 수 있다. 즉, 기존 플로들의 전송률이 감소한 상태에서 이동단말 플로가 데이터를 전송할 수 있다. 반면에 제안한 포워딩 제어 방식을 적용한 경우에는 버퍼링된 이동단말의 패킷들을 추정된 혼잡 상태에 따라 폐기하므로 기존 플로들에게 미치는 영향이 감소하게 된다. 즉, 무조건 포워딩하는 방식에 비해 기존 플로들의 전송률 감소가 작은 상태에서 이동단말 플로가 데이터를 전송하게 된다. 따라서 그림 13의 결과와 같이 제안한 포워딩 제어 방안을 적용한 경우가 무조건 포워딩하는 경

우보다 작거나 동등한 이동단말의 평균 수율을 나타낼 수 있다.

IV. 결 론

본 논문은 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장의 스무스 핸드오프 방식으로 이동단말이 혼잡이 존재하는 라우터와 연결된 새로운 서브네트워크로 이동한 경우, 기지국 패킷 버퍼링 방식이 결합된 모바일 IP 경로 최적화 확장이 미치는 영향을 분석하고 새로운 서브 네트워크와 연결된 라우터가 사용하는 RED 버퍼 관리 방식의 혼잡 상태에 따라 이전 기지국이 저장한 패킷들을 포워딩하거나 폐기하는 패킷 포워딩 제어 방안을 제안하였다. 시뮬레이션 결과로부터 제안한 패킷 포워딩 제어 방안은 이동 네트워크의 혼잡한 링크를 경유하는 TCP 플로들의 링크 이용률을 증가시킬 수 있으나 이동단말 플로의 평균 수율은 증가시키지 못하였다. 따라서 이동단말 플로의 수율과 경유하는 TCP 플로들의 링크 이용률을 동시에 증가시킬 수 있는 추가적인 연구가 필요하다고 사료된다.

참 고 문 헌

- [1] C. E. Perkins, "IP Mobility Support for IPv4," *revised draft-ietf-mobileip-rfc2002-bis-03.txt*, 2000.
- [2] C. E. Perkins, "Mobile IP," *International Journal of Communication Systems*, pp.3-20, 1998.
- [3] C. E. Perkins, "Mobile IP," *IEEE Communications Magazine*, pp. 84-99, May 1997.
- [4] C. E. Perkins and D. E. Johnson, "Route Optimization in Mobile IP," *draft-ietf-mobileip-optim-10.txt*, 2000.
- [5] R. Caceres and L. Iftode, "Improving the Performance of Reliable Transport Protocols in Mobile Computing Environments," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, pp. 100-109, Nov. 1995.
- [6] Doo Seop Eom, and et. al., Performance Improvement by Packet Buffering in Mobile IP Based Networks, *IEICE Transactions on communications*, vol. E83-B, no. 11, pp. 2501-2512, Nov. 2000.

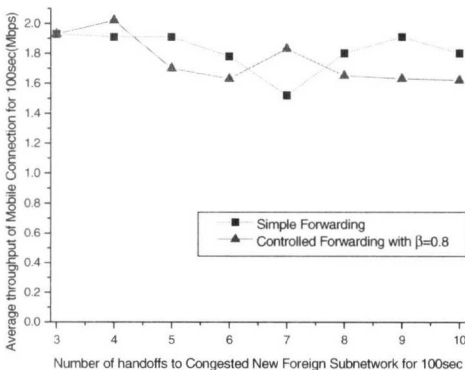


그림 13. 100초간 혼잡한 서브 네트워크로 이동한 회수에 따른 이동 단말의 평균 수율 성능

[7] R. Caceres and V. Padmanabhan, "Fast and Scalable Handoffs for Wireless Networks," *Proceedings of the ACM Mobicom'96*, pp. 56-66, Nov. 1996.

[8] H. Balakrishnan, and et. al., A Comparison of Mechanisms for Improving TCP Performance over Wireless Links, *IEEE/ACM Trans. Networking*, vol.5, no.6, pp.756-769, Dec. 1997.

[9] S. Floyd, V. Jacobson, Random Early Detection gateways for Congestion Avoidance, *IEEE/ACM Transactions on Networking*, vol. 1, no.4, pp. 397-413, August 1993.

[10] Doo Seop Eom, and et. al., Improving TCP Handoff Performance in Mobile IP Based Networks, to be published *Computer Communications*, vol. 25/7, pp. 635-646, April 2002.

[11] ns-2 SIMULATOR : <http://www.isi.edu/nsnam/ns/>, 2001

허 경(Kyeong Hur)

정회원

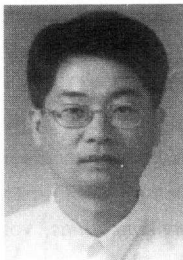


1998년 2월: 고려대학교 전자공학과 학사
 2000년 2월: 고려대학교 전자공학과 석사
 2000년 3월~현재: 고려대학교 전자공학과 박사과정 재학 중

<주관심 분야> 통신네트워크 설계 및 성능분석,
 IP 네트워크, 이동 멀티미디어 시스템

엄 두 섭(Doo-Seop Eom)

정회원



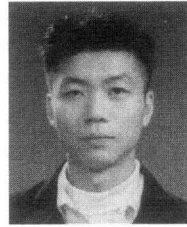
1987년 2월: 고려대학교 전자공학과 학사
 1989년 2월: 고려대학교 전자공학과 석사
 1999년 3월: 일본오사카대학 정보통신공학과 박사
 1989년 2월~1999년 8월: 한국전자통신연구소 연구원

1999년 9월~2000년 8월: 원광대학교 전임강사
 2000년 9월~현재: 고려대학교 전기전자전파공학부 조교수

<주관심 분야> 통신네트워크 설계 및 성능분석,
 무선 ATM, IP 네트워크

이 승 현(Seung Hyun Lee)

정회원



2001년 2월: 고려대학교 전자공학과 학사
 2001년 3월~현재: 고려대학교 전자공학과 석사과정 재학 중

<주관심 분야> 통신네트워크 설계 및 성능분석

차 균 현(Kyun Hyon Tchah)

정회원



1965년 2월: 서울대학교 전기공학과 학사
 1967년 6월: 미국 일리노이 공과대학 석사
 1976년 6월 : 서울대학교 전자공학과 박사
 1977년 3월~현재: 고려대학교 전자공학과 교수

1998년 1월~1998년 12월: 한국통신학회 회장
 1998년 4월~현재: 한국전자통신연구원 부이사장
 <주관심 분야> 통신 이론, 이동 통신, 위성 통신,
 이동 멀티미디어 시스템