

이종망간의 핸드오프에서의 Freeze-TCP 성능 저하 개선방안

An Enhancement Method to Improve the Performance of the Freeze-TCP during Vertical Handoff between Heterogeneous Networks

전 성 우

(광운대학교 전자통신학과, 석사과정)

박 영 충

(전자부품연구원)

민 상 원

(광운대학교 전자통신학과, 교수)

정 광 모

(전자부품연구원)

Key Words : Freeze-TCP, Duplicate ACK, RTO, Persist mode, Heterogeneous Networks

목 차

I. 서론	1. 문제점
II. 핸드오프로 인한 TCP 성능 저하	2. 제안한 해결방안
III. Freeze-TCP	VI. 결론 및 향후 과제
IV. Duplicate ACK	참고문헌
V. 이종망간 핸드오프에서의 문제점과 해결방안	

I. 서론

최근 이동통신기술의 향상과 인터넷 서비스의 증가는 TCP/IP 프로토콜에 대한 변화를 초래하고 있다. 이전의 유선 망만을 고려하여 만들어진 TCP/IP 프로토콜은 인터넷 서비스의 이동성 지원과 이동통신망에서의 인터넷 서비스 지원을 통해서 무선망에서의 TCP/IP가 활용되고 있다. IP에서는 이미 이동성을 고려하여 Mobile IP에 대한 연구가 많이 진행되어 활용되고 있다. TCP 또한 IP와 함께 이동망에서 사용되고 있는데 이런 과정에서 많은 문제점들이 나타나고 있다. TCP는 기본적으로 유선망을 기반으로 하여 만들어진 프로토콜이다. 패킷 손실에 대해서는 혼잡에 의한 손실로만 가정하며 혼잡 제어 메커니즘을 통해서 패킷 손실을 조절한다. 그러나 무선망에서는 패킷 손실을 일으키는 원인이 혼잡에 의한 것만이 아니다. 무선망은 유선망에 비해서 대역폭도 제한적이며 에러 발생률도 현저히 높다. 높은 에러 발생률은 무선망의 특징으로 이미 알려져 있는 사실이다. 또한 무선망에서는 이동성 지원시 발생하는 핸드오프나 주변 환경에 의한 fading에 의한 끊김이 발생한다. 이런 연결 끊김 현상은 패킷 손실의 결정적인 영향을 미친다.[1]

무선 환경에서의 TCP 적용에 대한 문제점을 해결하기 위한 여러 가지 방법들이 이미 제시되어 있다. 그 중에 핸드오프에 의한 TCP 성능 저하 문제를 해결하기 위한 방안으로 Freeze-TCP 방법이 있다. Freeze-TCP는 TCP의 기본 원리인 end-to-end 방식을 고려한 TCP 성능 개선 방법이다. 본 논문에서는 핸드오프를 통한 TCP 성능 저하 개선방안으로 Freeze-TCP를 사용하여 이종망에서 발생하는 문제점을 해결하고 Duplicate ACK에 대해서도 함께 해결하고자 한다.

II. 핸드오프로 인한 TCP 성능 저하

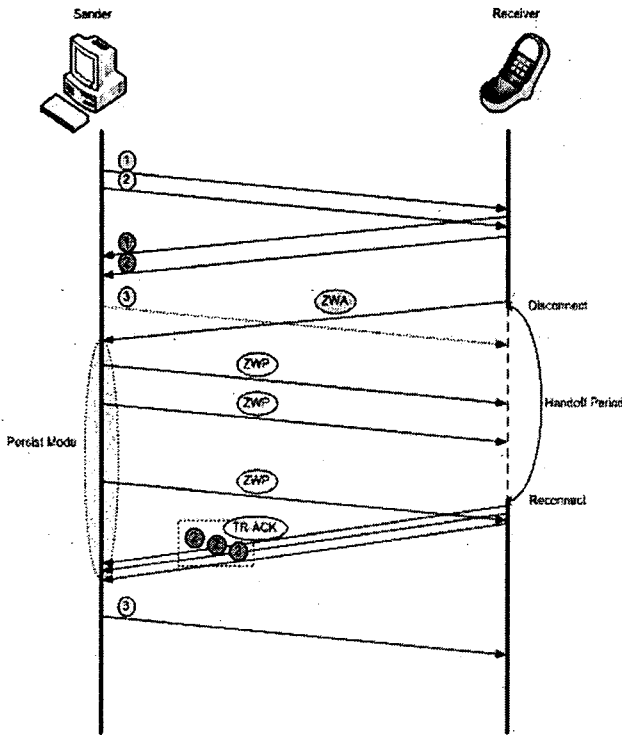
무선망에서 핸드오프를 할 경우에 TCP에서 발생하는 문제점으로 우선 패킷 손실을 생각할 수 있다. 핸드오프 동안에는 두 end host가 disconnection 상태이기 때문에 패킷을 잃어버리게 된다. 패킷 손실이 발생하면 패킷 송신측 TCP에서는 전송한 패킷에 대한 응답을 받지 못하게 되고 retransmission timeout이 발생한다. Retransmission timeout은 전송된 패킷에 대한 응답 가능 시간을 RTT(Round trip time)를 기반으로 정하여 패킷 전송시 설정하여 보낸다. Retransmission timeout은 전송한 패킷에 대한 응답 가능 시간을 나타낸 것으로 Retransmission timeout이 발생하면 송신측 TCP는 혼잡이 발생한 것으로 간주하고 slow start 메커니즘을 실행한다. Slow start는 혼잡제어 방법 중 하나로써 window size를 1로 초기화하여 데이터를 전송하는 것을 말한다. 대역폭을 활용을 최대화하고 혼잡을 막기 위해서 TCP는 window size를 조금씩 늘려가는 방식으로 데이터를 전송한다. Slow start는 하나의 window 크기의 데이터를 전송하고 응답 메시지를 받으면 2배씩 window size를 증가시켜 데이터를 보내게 된다. 핸드오프에서 고려해야 할 문제점이 증가된 window size가 핸드오프에 의한 Retransmission timeout으로 초기화 된다는 것이다. Retransmission timeout이 발생하면 이어서 slow start가 실행된다. 그런데 Retransmission timeout과 slow start이 처음 유선망에서의 혼잡 제어를 위한 것이기 때문에 무선망에서의 잦은 핸드오프에 의한 disconnection으로 retransmission time발생에 대한 해결책으로 slow start를 쓰는 것은 TCP의 성능에 큰 지장을 초래한다. [2][3]

Freeze-TCP는 핸드오프로 인한 disconnection으로

retransmission timeout이 발생하여도 slow start에 의한 window size 초기화가 이루어 지지 않고 retransmission timeout 값이 급격히 증가되는 것을 막아준다.

III. Freeze-TCP

Freeze-TCP는 end-to-end 방식을 유지한 무선망에서의 TCP 성능 향상을 위한 방법이다. Freeze-TCP의 가장 큰 특징은 핸드오프 기간동안 송신측 TCP가 persist mode로 변환되는 것이다. Persist mode에 있는 동안은 retransmission timeout이 진행되지 않고 멈추게 된다. 그리고 핸드오프 이전의 congestion window size를 저장하였다가 핸드오프가 종료되고 재연결이 설정된 후에 이전의 congestion window (CWIN) size 값을 계속 유지하여 데이터를 전송한다.



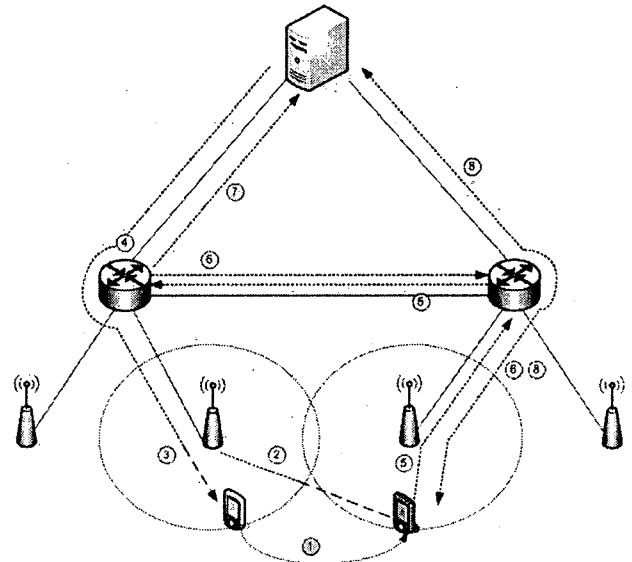
<그림 1> Freeze-TCP 동작 과정

그림 1은 Freeze-TCP에서의 동작 과정을 보여주고 있다. Freeze-TCP의 주된 동작 과정은 이동중인 수신자가 신호의 강도를 통하여 disconnection이 발생할 것을 예측하여 Zero window advertisement(ZWA) 메시지를 송신자에게 보내면서 시작된다. 수신자로부터 ZWA 메시지를 받은 송신자는 현재 진행중인 데이터 전송을 멈춘다. 그리고 현재 사용중인 congestion window size 값을 저장하고 retransmission timeout 값을 멈추게 된다. 핸드오프가 종료되면 수신측에서는 재연결을 설정하고 핸드오프가 종료된 것을 송신자에게 알리기 위해 핸드오프 이전에 받은 최종 메시지에 대한 응답 메시지를 3번 연속으로 보내게 된다. 이것을 Triplicate ACK(TR-ACK)라고 한다. TR-ACK를 받은 송신자는 persist

mode에서 벗어나면서 persist mode로 변화되기 전에 저장해 둔 CWIN 값과 이전의 RTO 값으로 데이터를 전송한다. 그래서 핸드오프를 통한 TCP 성능의 가장 큰 단점인 핸드오프 후의 slow start에 의한 CWIN의 초기화의 문제점을 해결할 수 있다.[4] 이동망에서의 핸드오프는 자주 발생하게 되며 핸드오프에 의한 잦은 CWIN 값의 초기화가 발생하게 되면 대역폭에 대한 제한으로 무선망에서의 데이터 전송에 대한 부담이 증가되며 효율적인 throughput을 제공하지 못하게 된다. Freeze-TCP는 핸드오프로 인한 TCP 성능 개선을 위한 좋은 해결책이 된다.

IV. Duplicate ACK

핸드오프를 통해 TCP에서 발생할 수 있는 문제점으로 duplicate ACK가 있다. 이것은 송신자가 전송한 데이터가 핸드오프에 의한 지연으로 응답을 늦게 수신함으로써 발생하게 된다.



<그림 2> 핸드오프 시 데이터 전달 과정

그림 2는 네트워크 계층에서의 핸드오프 시 패킷 전달 과정을 보여주고 있다.

1. 이동 단말의 이동으로 인한 핸드오프 발생.
2. 이동 단말의 이동으로 이전 기지국과의 연결이 끊김.
3. 이동 단말의 이동으로 인해서 이동 전 이동 단말에 전송중인 패킷 손실 발생.
4. 핸드오프로 인해 이동단말에 전송되던 데이터를 저장.
5. 새로운 도메인으로 이동한 이동 단말이 binding update 메시지를 Home Agent(HA)에 전달하여 이동 사실을 알림.
6. HA에서 저장해 둔 패킷을 이동노드에게 터널링으로 전달.
7. HA는 Correspond Node(CN)에게 이동노드의 위치 변경을 알림.
8. CN은 HA를 거치지 않고 이동노드에게 직접 데이터를 전송.

패킷 손실은 핸드오프 시 3번 과정에서만 발생하게 된다. 핸드오프 시 전송되는 ACK로는 핸드오프 이전에 받은 데이터에 대해서 핸드오프 후에 ACK를 보내주는 경우와 핸드오프 후 HA에 저장된 데이터를 받고 ACK를 전달하는 두가지 경우를 생각할 수 있다. Duplicative ACK는 핸드오프 종료 후 재전송된 데이터에 대한 ACK와 핸드오프 이전에 전송된 데이터에 대한 뒤늦은 ACK에 대한 중복 ACK 문제이다.[5]

V. 이종망간 핸드오프

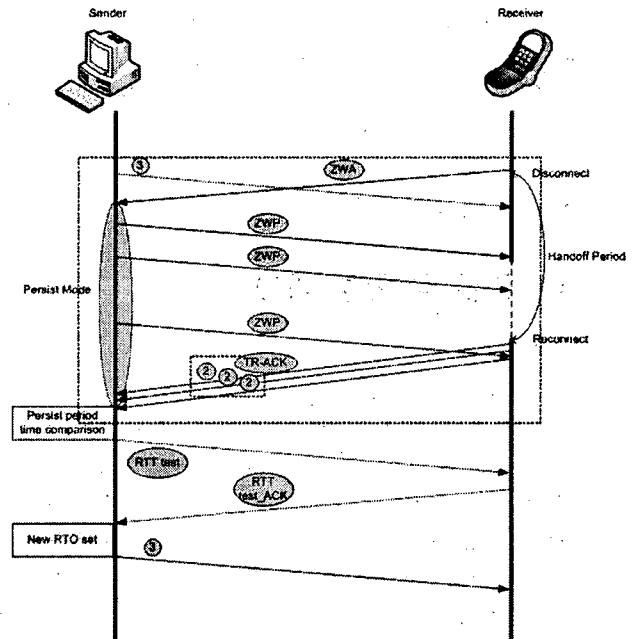
1. 문제점

유무선망의 통합화를 통해서 무선망에서의 TCP/IP의 적용에 대한 문제와 해결방안들이 제시되고 실행되고 있다. 유무선망의 통합은 현재 많은 이종망간의 이동성 지원을 요구하고 있다. All-IP 망을 통한 통합망을 향해 나아가고 있는 현재에 이종망간의 원활한 연결과 seamless한 데이터 전송이 필요하다.

이종망간의 핸드오프에 있어서 TCP의 성능 저하 또한 해결해야 할 문제이다. 이종망간의 핸드오프는 동일망간의 핸드오프와 다른 특징을 가진다. 이종망간의 핸드오프에는 인증관련 절차가 추가되어서 동일망간 핸드오프에 비해 더 긴 전송 지연이 발생하게 된다. 이것은 핸드오프로 인한 지연 시간이 길어지게 되며 seamless한 데이터 전송에 지장을 초래할 수 있다. 이러한 이종망간 핸드오프는 TCP 성능에도 영향을 미치는 데 이종망간의 핸드오프에서 발생할 수 있는 가장 큰 특징으로 RTT의 변화를 생각할 수 있다. 서로 다른 이종망에서는 각 망의 특징이 확연히 구별되어 나타난다. 특히 데이터 전송에 대한 RTT 값의 차이는 TCP의 성능 저하를 일으킬 수 있다. WLAN과 같은 망에서 데이터 서비스를 지원하는 이동통신망으로 이동시 망 자체내 환경의 차이로 인해서 RTT값의 큰 변화가 일어날 수 있다. 넓은 대역폭과 빠른 전송률을 가진 망에서 보다 한정된 대역폭과 보다 느린 전송률을 가진 망으로의 핸드오프가 발생했을 경우에 Freeze-TCP를 사용하는데 있어서 문제가 발생한다. Freeze-TCP의 가장 큰 특징인 핸드오프로 인한 CWIN값의 초기화 문제가 다시 발생할 수 있다. 이동단말이 RTT값이 핸드오프 이전망의 RTT값에 비해서 RTT값이 큰 망으로 이동하였을 경우를 생각해 보자. Freeze-TCP를 통해서 핸드오프 종료후 데이터를 재전송할 경우에 송신자는 이전 RTO값을 그대로 사용하여 데이터를 전송하게 된다. 그러나 이전 RTO값은 이전 망의 RTT값을 고려하여 생성된 값이다. 핸드오프 이후의 망이 이전 망보다 RTT값이 현저히 크다면 핸드오프 이후 전송하는 데이터들은 대부분 retransmission timeout이 발생하게 될 것이다. 이것은 Freeze-TCP를 통해서 CWIN 값이 초기화 되는 것을 막음으로 throughput이 떨어지는 문제를 해결한 것을 다시 retransmission timeout 발생을 통해서 CWIN 값의 초기화를 유발하여 throughput 저하를 초래하게 된다.

2. 해결방안

본 논문에서는 이런 문제를 persist period time table과 RTT test 패킷을 통해서 해결하고자 한다. 그리고 이 제안 방법은 duplicate ACK에 대한 문제도 해결해 준다.



<그림 3> 제안한 방법의 활용 과정

송신자는 수신자의 ZWA와 TR-ACK를 통해서 핸드오프에 대한 성능 저하를 위해 persist mode로 변환된다. TCP timer에는 persist timer 라고 해서 TCP에서는 persist mode에서 영구적으로 있는걸 방지하기 위해서 persist timer를 사용한다. Persist timer는 persist mode로 변환된 후 상대방의 연결이 끊기거나 재연결이 불가능한 경우를 대비해서 특정 시간이 지나면 연결을 끊기 위해 사용된다. 이것과는 달리 Freeze-TCP에서는 ZWA와 TR-ACK 사이의 시간을 핸드오프 시간으로 간주할 수 있다. 송신자는 ZWA와 TR-ACK 사이의 persist period time 값을 저장하여 매번 persist mode로 전환 될 경우 이전 persist period time값과 비교한다. 동일망에서의 핸드오프로 인한 persist period time값을 비슷한 시간 값을 갖겠지만 이종망간의 핸드오프는 인증관련 문제들로 인해 보다 긴 persist period time 값을 가질 것이다. 이것으로 이종망간의 핸드오프를 예측할 수 있다. 그리고 이종망간의 핸드오프를 예측한 후에는 이동한 망의 RTT값의 차이에 따른 retransmission timeout 발생을 막기 위해서 이동망의 RTT값을 기반으로 한 RTO값의 재설정이 필요하다. 이에 대한 해결책으로 window size 1의 RTT 측정을 위한 패킷인 RTT test 메시지를 전송한다. 이 과정에서 새로운 망의 RTT 값의 측정과 함께 duplicate ACK에 대한 문제도 해결할 수 있다. 핸드오프 발생 전에 수신한 데이터에 대해서 수신자는 TR-ACK 메시지 전송과 함께 ACK를 전송할 수 있다. 그리고 핸드오프 후의 재설정 후에 받은 메시지에 대해서는 duplicate ACK를 유발할 수 있으나 RTT test 메시지 전송이 송신자에서 수신자로의 왕복 패킷 전달 수신과정을 거치기 때문에 이 시간 동안에 duplicate ACK

가 가능한 데이터에 대한 응답을 받을 수 있다. RTT test 패킷 전송 및 응답 기간 동안은 핸드오프 종료후의 데이터 패킷 전송이 이루어 지지 않았기 때문에 duplicate ACK에 대한 문제 고려를 하지 않아도 된다. RTT test 메시지를 통해서 이동한 망의 RTT 값을 구하면 이에 대한 RTO 값을 설정하여 데이터를 전송하게 된다. RTT test 값에 대한 응답을 받게되면 핸드오프 이전에 전송한 패킷에 대해서 손실된 패킷을 제외한 패킷의 응답을 받게 된다.

이러한 과정을 통해서 Freeze-TCP의 성능 저하와 duplicate-ACK 문제를 해결할 수 있다.

V. 결론 및 향후 과제

최근 네트워크 통합화에 대해서 이슈가 되고 있는가운데 유무선 통합과 이종망간의 연결에 대한 연구가 활발히 진행되고 있다. 그 가운데 이종망간의 핸드오프에서의 TCP 성능 저하 문제는 유선망을 기반으로 만들어진 TCP의 문제점을 더 추가시켰다. 핸드오프를 통한 패킷 손실에 대한 TCP 문제의 해결책인 Freeze-TCP에 대해서 persist period time값의 비교를 통한 동일망과 이종망간 핸드오프를 구별하고 새로운 망에서의 RTT값을 통해서 최적의 RTO값을 설정하여 retransmission timeout 발생을 막으며 핸드오프를 통한 duplicate ACK 문제를 해결하였다.

향후 이에 대한 시뮬레이션을 이용하여 이전 방법들과의 지연 및 throughput을 비교를 함으로써 본 제안을 통한 성능 향상을 확인하고자 한다.[6]

참고문헌

1. Y. Tian, et al., "TCP in Wireless Environments: Problems and Solutions", IEEE Radio Communications, March 2005.
2. W. Stallings, "High-Speed networks and Internets", Prentice Hall, 2001.
3. M. Hassan and R. Jain, "High Performance TCP/IP Networking", Person Prentice Hall, 2004.
4. T. Goff, et al., "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments", IEEE INFOCOM, 2000.
5. H. Soliman, "Mobile IPv6", Addison Wesley, 2004.
The Network Simulator- ns-2, <http://www.isi.edu/nsnam/ns/index.html> Stephen A. Thomas, IPng and the TCP/IP Protocols,, Wiley, 1996.