

M-TCP 전송에서 링크 이용율에 따른 트래픽 처리

Traffic Throughput on based Link Available in M-TCP Transfer

송선희^o · 석경휴 · 김광준^ㄴ · 김문환^ㄹ · 박동석 · 배철수^ㄷ · 나상동
조선대학교, ^여수대학교, ^한국전파기지국, ^관동대학교
Sun-hee Song · Kyung-hyu Suk · Kwan-Jun Kim · Moon-hwan Kim^ㄹ
Dong-suk Park · Chul-soo Bae^ㄷ · Sang-dong Ra
Dept. of Computer Engineering, Chosun University
^oDept of Computer Engineering, Yosu National University
^ㄴKRTnet Corporation Network Planning
^ㄹDept. of Field of Com. Science, Kwandong University
E-mail: ssh6061@nate.com

요 약

무선네트워크 TCP는 패킷손실을 혼잡에 의한 것으로 보고 전송률을 감소시키는데, 고대역폭 연결의 효율성을 높이기 위해 M-TCP에서의 링크이용율에 따른 트래픽 처리를 연구한다. M-TCP는 무선네트워크에서 연결 중단이 자주 발생하고 동적으로 변하는 대역폭으로 인한 비트율이 낮은 무선 링크에 효과적인 프로토콜이다. 연결 중단 시 송신노드가 인접한 경우나 멀리 떨어져 있는 경우 유리하며, M-TCP가 링크 계층 솔루션과 함께 사용될 경우 TCP 성능이 향상된다. 본 논문에서는 백그라운드 트래픽 흐름은 그대로 유지하여 M-TCP상에서 연결의 개수를 늘린 후 그 결과 발생하는 대역폭 공유 반응 및 링크 이용율로 M-TCP 연결로부터 생긴 트래픽 처리를 분석한다.

키워드

무선네트워크, 혼잡제어, M-TCP

I. 서 론

무선네트워크 TCP[1]는 신뢰성 있는 엔드 투 엔드 전송 계층 프로토콜 윈도우 기반 패킷손실 및 재전송 타임아웃에 따라 전송률이 결정된다.

TCP에서는 모든 패킷 손실이 정체로 인식되기 때문에 윈도우 크기가 줄어들고 전송률이 감소한다. 또한 왕복 지연 시간 증가로 인한 재전송 타임아웃이 늘어난다. TCP는 유선네트워크를 위해 설계되었기 때문에 무선네트워크의 성능을 높이기 위한 최적화된 전송 프로토콜[2]의 다양한 제안이 제시되고 있다. 링크상에서 송신되는 문자나 프레임의 양을 적절히 조절하여 수신측에서 항상 충분한 버퍼를 유지하도록 하는 역할을 수행한다. 무선 네트워크에서 고대역폭 연결의 효율성은 저대역폭 윈도우 전송 시간에 의해 제한되며, 이것은 심각한 성능의 감소를 초래한다.

M-TCP는 게이트웨이 노드의 처리 시간이 짧을 경우 확장 가능하지만, 이동 호스트 배터리의 수명을 크게 연장시키지는 않는다. 이 프로토콜에서 BER이 낮다는 가정은 BER이 10^{-5} 보다 악화되지 않음을 입증한 여러 연구를 통해 뒷받침되는데, 링크 계층에서 적절한 순방향 에러 정정 구

조 및 재전송 구조를 사용할 경우 그 비율은 더 낮출 수 있다.

본 논문에서는 동일한 중단 노드에서 연결의 승인을 가지고 있는 명확한 무선 네트워크의 가상 채널을 통해 전송된 트래픽의 효율성과 백그라운드 트래픽 흐름은 그대로 유지하고 M-TCP 상에서 연결 개수를 늘린 후 그 결과 발생하는 대역폭 공유 반응을 연구한다. 출력 IP 큐의 승인, 무선 네트워크 링크 계층에서의 출력 트래픽, 대역폭 지연의 지터처리에 대해 시뮬레이션 한다.

II. M-TCP 백그라운드 트래픽 처리

2.1. M-TCP 스택

M-TCP(Mobile-TCP)[3]는 비트 오류나 연결 중단으로 인해 패킷이 손실될 때 송신노드가 정체 윈도우를 줄이지 않도록 막아 무선 네트워크에서의 TCP 성능을 높인다. 이동 호스트는 송신노드가 타임아웃하고 재전송할 때까지 기다리지 않고 재연결 즉시 데이터를 받기 시작하며, 연결 중단이 자주 발생하고 동적으로 변하는 대역폭 때문에 비트율이 낮은 무선 링크에서 효과를 낼 수 있다. 그림 2-1에서와 같이 프로토콜은 이동 호스

트가 가장 낮은 레벨인 3계층 구조를 지원한다. 셀 레벨은 이동 지원국의 최상위 레벨에는 관리 호스트(SH)가 있고 관리 호스트 아래의 도메인별로 그룹화하여 통제한다. 이동 지원국은 연결 중단점 역할만 수행하는 간단하고 저렴한 장치로 대부분의 라우팅, 대역폭 관리, 이동 호스트 추적, 로컬 오류 복구 및 기타 세부적인 프로토콜 사항을 관장한다. 또 관리 호스트는 유선 네트워크에 연결되어 게이트웨이 역할을 하면서 송신노드와의 TCP 연결 및 이동 호스트(MH)와의 M-TCP 연결을 유지하며, 관리 호스트-TCP 클라이언트는 이동 호스트로 전달될 ACK를 송신노드에서 M-TCP로 전송한다. M-TCP 클라이언트 역시 송신노드로 전달될 ACK를 이동 호스트에서 관리 호스트-TCP로 전송한다.

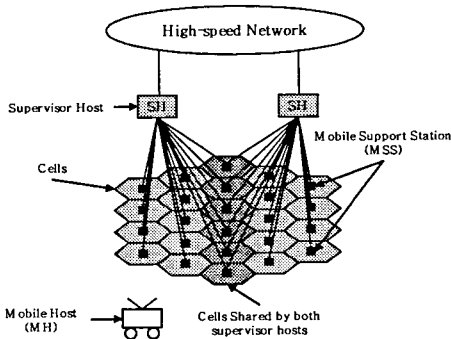


그림 2-1. 이동 TCP 구조
Fig. 2-1 Mobile TCP architecture

또 관리 호스트는 RTT(Retransmission Timeout Timer)를 유지 관리하는데, 이동 호스트로부터 ACK를 받는 즉시 관리 호스트는 마지막 바이트를 제외하고 모든 ACK를 송신노드로 보낸다. RTO 타이머가 만료되면 연결 중단, 낮은 대역폭, 시그널 페이딩으로 인해 이동 호스트로부터 새로운 ACK는 수신되지 않으며 관리 호스트는 송신노드로 마지막 바이트의 ACK를 보낸다. 이 ACK는 0으로 업데이트된 송신노드용 윈도우 크기 정보를 포함하고 있으며 송신노드가 지속 모드에 들어가게 한다. 이 상태에서는 재전송 타임아웃의 영향을 받지 않으며 재전송 타이머를 지수 백오프시키지 않고 정제 윈도우를 줄이지 않는다.

또 호스트가 연결되지 않았고 가용 대역폭이 거의 없는 셀에 머물러 있다면 관리 호스트-TCP는 마지막 바이트의 ACK에서 윈도우 크기를 0으로 설정하여 송신노드 윈도우를 미리 축소시키는 방법으로 송신노드의 지수 백오프를 방지한다. 재연결된 이동 호스트는 환영 패킷을 관리 호스트로 보내고, 관리 호스트는 ACK가 없는 모든 패킷을 이동 호스트로 재전송하고 송신노드에게 전송 윈도우를 완전히 열어주도록 요청한다. 핸드오프는 관리 호스트 레벨 도메인 사이를 이

동할 때 발생하는데, 핸드오프가 발생하면 TCP 발신 윈도우는 0으로 줄어들고 일부 상태 정보는 일련화되어 새로운 관리 호스트로 전달되며 소켓 비퍼의 내용은 폐기되고, 새로운 관리 호스트는 발신 윈도우를 원래의 크기로 다시 열게 된다.

관리 호스트의 대역폭 관리 모듈은 각 연결에 대해 대역폭을 할당, 모니터링, 재할당하고 지정된 속도로 데이터가 전송되는지 확인하면서 QoS를 이행한다.

2.2 M-TCP 흐름제어

양방향 트래픽 하에서 M-TCP상의 연결개수를 늘려 트래픽 흐름과 효율성을 살펴보고자 각기 노드에서 출력 IP 큐의 최대 크기에 대한 식을 유도한다. 노드 j에서 최대 큐 크기는 연결 I의 번잡 주기의 첫 번째 세그먼트가 노드 j에 도착할 때 알 수 있다. 큐 크기는 노드에서 발생하는 연결의 윈도우 크기를 절대로 초과하지는 않는다. 안정적인 상태에서 양방향 트래픽하의 노드 I의 IP큐의 최대 점유기간 Q_i max는 식(2.1)와 같다.

$$Q_i \max \leq \begin{cases} W_i - (L_{ij} + L_{ji}), & W_i > W_j a + (L_{ij} + L_{ji}) \\ (W_i + W_j) - (L_{ij} + L_{ji}), & W_i - (L_{ij} + L_{ji}) \leq W_j \leq W_i + (L_{ij} + L_{ji}) \\ W_i & \text{otherwise} \end{cases} \quad (2.1)$$

두 번째 항은 직접적으로 식(2.2)의 결과로부터 발생하고 세 번째 경우는 큐 크기가 노드에서 발생하는 연결의 윈도우 크기를 초과할 수 없다는 것이다.

$$W_j \cdot (L_{ij} + L_{ji}) \leq W_j \leq W_j + (L_{ij} + L_{ji}) \quad (2.2)$$

첫 번째 항을 고려하면 노드 i에서 최대 큐 크기는 연결 j의 번잡 주기의 첫 번째 세그먼트가 노드에 도착할 때 알 수 있다. $\tau_{i,m}$ 은 연결 j의 m번째 번잡 주기의 첫 번째 세그먼트가 노드 i에 도착할 때 시간이다. 이러한 시간에서 큐 크기는 노드 i에서 큐된 세그먼트의 수에 의해 할 수 있고 네트워크의 나머지에서 연결 i에 속해 있는 세그먼트 수와 승인 수를 더한 것은 윈도우 크기 W_i 와 같다.

연결 j의 m번째 번잡 주기의 첫 번째 세그먼트가 $t_1 = \tau_{i,m} \cdot D_{ji}$ 시간에 노드 j에 의해서 전송된다. t_1 시간에 노드 j의 출력 큐는 연결 i의 어떠한 승인도 포함하지는 않는다. $(t_1, \tau_{i,m})$ 간격 동안에 채널 i로부터 전체적인 세그먼트 $(t_1, \tau_{i,m}) = D_{ji} \rho = L_{ji}$ 는 노드 j에 도착된다. 더욱이 연결 I의 L_{ij} 는 노드 j에 도착된다. 더욱이 연결 I의 L_{ij} 세그먼트는 $\tau_{i,m}$ 시간에 노드 i로부터 노드 j까지 링크 상에서 전송된다. 그러므로 $\tau_{i,m}$ 에서 전체적인 시스템에서 연결 i의 전체적인 세그먼트와 승인은 식(2.3)과 같다.

$$Q_i(\tau_{i,m}) + L_{ij} + L_{ji} \quad (2.3)$$

윈도크기 W_i 와 식(2.4)를 같게하면 식(2.4)와 같다.

$$Q(\tau_{j,m}) = W_i(L_{ij} + L_{ji}) \quad (2.4)$$

양방향 배치 구조에서 전송율이 다른 보다 일반적인 비대칭적인 경우로 확장하여 적용하고 ρ_i 와 ρ_j 를 각각 연결 i 와 j 의 전송율을 나타내면 승인 압축의 생성에 대한 조건은 식(2.5)와 같다.

$$\frac{w_i + W_j}{\rho_i + \rho_j} > D_{ij} + D_{ji} \quad (2.5)$$

즉 두 연결의 윈도우의 전송 시간의 합은 발생시키는 승인 압축에 대한 라운드 전송지연보다 반드시 크다.

2.3 네트워크 구성 및 시뮬레이션 결과

본 논문에서의 시뮬레이션 환경 설정은 네트워크 시뮬레이터 ns-2를 기반으로 하는 시뮬레이션을 환경을 적용하였다. 전송률 기반 확장 TCP Rate에 M-TCP 프로토콜에 추가해 그림 2-2에서와 같이 서버 2대, 클라이언트 n 개, $n \geq 33$ 의 네트워크 배치 구조에서 게이트웨이 G_1 과 G_2 를 병목 링크되도록 연결하였다. 링크 대역폭은 10Mbps로 설정하였으며 각 링크의 대기 시간은 5ms이다. 최대 세그먼트 크기는 1kb로 고정되어서 일부 클라이언트, 즉 32개의 연결은 다른 연결의 백그라운드 트래픽 역할을 하며 트래픽 비중이 길게 늘어진 크기의 파일들을 병목 링크를 통해 서버로 양방향 전송하였다. 연속 전송 사이의 지수 시간에는 유휴 상태를 유지하면서 성능을 측정하기 위한 연결은 항상 전송할 데이터가 있는 무한 소스이며 신뢰 처리율을 최대화하기 위해 M-TCP를 확장하여 실행하였다.

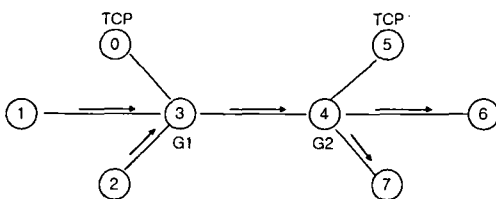


그림 2-2. 병목 링크 G_1, G_2 가 있는 네트워크 구성. left-to-right 트래픽 흐름
Fig.2-2 Network configuration with bottleneck link G_1, G_2 . Traffic from left to right

또 백그라운드 트래픽 흐름은 그대로 유지하고 M-TCP 상에서 연결의 개수를 늘린 후 그 결과 발생하는 대역폭 공유 반응에 대해 연구하였다. 대역폭, 버퍼 크기, 평균 파일 요청 크기, 기타 시스템 매개변수를 어떻게 지정하더라도 클라이언트 수 또는 연속파일 전송 간의 평균 유휴 시간을 조정함으로써 경쟁 레벨을 생성할 수 있다. 대역폭을 감안한 기본 성능 평가 설정의 일반적인

구성에서 처음 8개의 연결이 백그라운드 트래픽 역할을 하면서 클라이언트에서 서버로 패킷을 전송하는 것으로서 그림 2-3과 같이 나타낸다. TCP 백그라운드 트래픽에 대한 네트워크 정보는 시뮬레이션 정보, 현재 노드 정보 및 중단간 노드사이 시뮬레이션을 수행하기 위한 지연을 나타낸다. 또한 전체 네트워크에 대한 중간 노드의 평균 수신 패킷과 송신 패킷에 대한 정보 및 중간 노드에서 수행되는 시뮬레이션 처리 시간이다. 여기에서 63개의 패킷을 발생하여 송신함으로써 중간노드를 경유하여 전송된 포워드 패킷의 크기가 63개로서 정상적으로 전송되고 있음을 알 수 있었으며, 수신노드에서 수신된 수신 패킷의 크기가 59로서 3개의 패킷이 드롭되어 손실되어짐을 알 수 있다. 시뮬레이션 실행 시간은 26,397초 동안 지속되며 10ms 간격으로 추적기록을 수집해보면 트래픽 비중이 크고 길게 늘어진 샘플링을 볼 수 있다.

Simulation length in seconds	26397
Number of nodes	8
Number of generated packets	63
Number of sent packets	63
Number of forwarded packets	63
Number of received packets	59
Number of sent bytes	63000
Number of forwarded bytes	63000
Number of received bytes	2360
Maximal packet size	1000
Average packet size	535.7377

그림 2-3. TCP 백그라운드 트래픽 네트워크 정보
Fig.2-3 The network information of TCP background traffic

두 방향에서 전송율이 다르다면 고대역폭 연결의 효율성은 저대역폭 윈도우 전송 시간에 의해 제한되므로 심각한 성능의 감소를 초래한다. 이것을 설명하기 위해 윈도우 크기가 $W_i = W_j = 128$ Kbytes인 것을 선택하고 링크 지연들은 $D_{ij} = D_{ji} = 5$ ms인 예를 고려한다. 연결 j 가 155Mbps/s의 안정율로 할당된다.

그림 2-4는 연결 i 에 할당된 율의 함수로서 연결 i 와 j 의 효율성의 변동으로서 TCP 백그라운드 트래픽에 대한 왕복 지연의 지터에 대한 처리율을 나타내고 있다. 연결 i 에 대한 이용 가능한 대역폭이 낮을 때 연결 j 는 그것의 이용 가능한 링크 대역폭의 몇 퍼센트만이 이용할 수 있다. 연결 i 의 대역폭이 증가함으로써 연결 j 의 처리율은 약 70Mbps/s가 될 때까지 선형적으로 증가한 반면에 연결 i 의 처리율은 무의미한 값으로 유지된다. 이 지점을 넘어서 처리율은 방정식의 두 번째 경우에 의해서 결정되고 연결 j 에서 증가하는 처리율은 낮은 비율에서 발생한다. 연결 i 의 전송율이 70Mbps/s에서 155Mbps/s로 증가함으로써

55%에서 75%로 연결 j의 효율성이 증가하는데 이는 왕복지연 시간을 줄이는 결과를 초래한다. 이 영역에서 연결 i의 효율성은 연결 j의 효율성이 55%에서 75%로 증가한 것과 비교해서 왕복지연 시간에 대한 지터 처리율은 100%에서 50%로 감소한다.

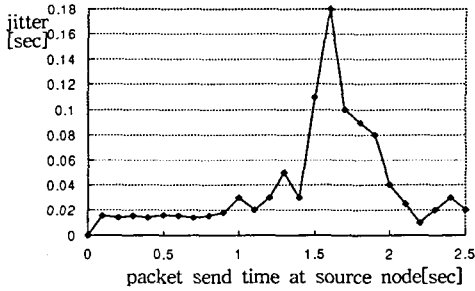


그림 2-4. 백그라운드 트래픽 지터의 처리율
Fig.2-4 Jitter throughput for background traffic

반대 연결의 데이터 세그먼트를 전송하기 위해 사용된 동일한 가상 채널을 공유하는 연결의 승인을 가지고 있는 명확한 무선 네트워크의 가상 채널을 통해 전송된 트래픽의 효율성을 이용하여 출력 IP 큐의 승인에 대한 서비스 시간을 아주 적게 설정하였으며, 출력 트래픽의 모든 큐잉은 무선 네트워크 링크 계층에서 발생하게 한다. 무선 네트워크의 링크 계층은 각각의 가상채널에 대해 분리된 큐를 제공하며, 각각의 가상 채널의 전송율은 51.6Mbps/s와 최대 윈도우 크기는 64Kbytes로서 7개의 세그먼트로 설정한 TCP 백그라운드 트래픽에 대한 중간노드의 포워드 패킷 처리율에 대한 시뮬레이션 결과를 나타낸 것으로서 그림 2-5와 같다.

중단 노드의 서로 다른 쌍 사이의 트래픽이 다른 가상채널을 통해 전송되는 것을 알 수 있으며, 상호간에 완전하게 분리되어 연결의 각 쌍에 대해 양방향 트래픽의 처리율을 이용하여 나타낸 중간노드의 왕복지연에 대한 처리율 결과를 나타냈으며, 이는 양방향 트래픽의 대역폭 지연의 값에 대한 TCP 연결 효율성이 지연이 없는 처리율을 보였다.

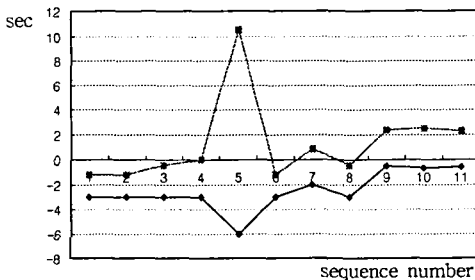


그림 2-5. TCP 백그라운드 트래픽에 대한 중간노드의 포워드패킷 처리율

Fig.2-5 Intermediate node forward packet throughput for TCP background traffic

III. 결론

M-TCP 전송에서 백그라운드 트래픽 흐름은 그대로 유지하고 M-TCP상에서 연결 개수를 늘린 후 그 결과 발생하는 대역폭 공유 반응에 대해 연구하였다.

동일한 중단 노드에서 연결의 승인을 가지고 있는 명확한 무선 네트워크의 가상 채널을 통해 전송된 트래픽의 효율성 및 무선 네트워크 링크 계층에서의 출력 트래픽, 대역폭 지연은 연결 중단 시 송신노드가 인접한 경우 TCP보다 M-TCP가 링크 이용율이 높았고, 송신노드가 멀리 떨어져 있는 경우에도 더 나은 효율을 보였다.

전송을 기반 혼잡제어 처리 M-TCP 연결로부터 생긴 트래픽을 분석한 결과 TCP 백그라운드 트래픽에 대한 처리율은 혼잡이 발생할 경우 왕복지연시간이 증가함을 알 수 있었으며, 중단 노드의 서로 다른 쌍 사이의 트래픽은 지연없는 처리율을 보였다.

참고문헌

- [1] H. Balakrishnan, S. Seshan, E. Amir and R. H. Katz, Improving TCP/IP Performance over Wireless Networks, in Proc. of ACM Mobicomm95, November 1995.
- [2] A. DeSimone, M. C. Chauh, and O. C. Yue, Throughput Performance of Transport Layer Protocol over Wireless LANs, Proc. GLOBECOM 93, December 1993.
- [3] Brown K. and Singh S., M-TCP: TCP for Mobile Cellular Networks, INFOCOM, 1996.
- [4] C. Lee, J. Jang, E.K. Park, S. Makkic. A simulation study of TCP performance over IEEE 1394 home networks. Computer Communications Vol. 26, pp. 670-678, 2003.