

# Ad-hoc 네트워크에서 다중경로를 지원하는 ATP 프로토콜에 대한 연구

이학주<sup>1</sup> · 장재신<sup>2\*</sup> · 이종협<sup>2</sup>

## A Study on a Multi-path ATP Protocol at Ad-hoc Networks

Hak Ju Lee · Jaeshin Jang · Jong Hyup Lee

### ABSTRACT

Wireless networks have several unique features : node mobility, restricted bandwidth, time-variable bandwidth, large latency, and high bit error rates time due to channel fading. These features at wireless networks intend to decrease the performance TCP protocols are used in wireless networks. Lots of studies have been done for finding appropriate wireless transport protocols for current wireless communications. However, related studies have not provided good performance or some protocols have a good performance only in specific circumstances. Thus, these are not suitable for general wireless circumstance. Therefore, we propose a new wireless transport protocol which provides better performance than the previous ones. And we'd like to solve a problem that previous protocols cannot maintain their connections even though they have multiple paths until another path is successfully set up. To solve these problems, a new protocol ATP-M is proposed which is designed on already known TCP-M and ATP protocols. With NS-2 computer simulation, it is shown that this newly proposed protocol has better system throughput than TCP, TCP-M and ATP protocols.

**Key words** : TCP, ATP, Multipath ATP-M, TCP-M, Transport Layer

### 요약

무선 네트워크의 특징들 중에 노드들은 이동성을 가지며, 제한된 대역폭과 대역폭의 변화, 높은 지연시간과 지연시간의 변화, 무선랜덤오류를 가지며 경로의 연결이 자주 단절된다. 이러한 무선 네트워크의 특징들이 무선 네트워크에서 유선 네트워크용 TCP를 사용 할 때에 성능을 떨어뜨리게 된다. 현재 무선 네트워크에 적절한 전송 프로토콜은 계속 연구 중이며, 기존에 연구된 연구 결과들은 성능이 우수하지 못하며, 특정한 환경에서만 좋은 성능을 보이기 때문에 일반적인 무선 네트워크 환경에는 적합하지 않다. 따라서 본 연구에서는 무선 네트워크에 알맞은 전송프로토콜을 연구하고 제안하기 위해 기존의 TCP 프로토콜과 이미 제안된 무선 전달계층용 프로토콜들을 분석한 뒤 성능이 더욱 우수한 무선 전달계층용 프로토콜을 제안하고자 한다. 또 다중경로를 사용하여 기존 연구가 경로의 연결이 끊기고 연결이 재설정될 때까지 전송을 할 수 없는 문제점도 해결한다. 이를 위해 TCP를 무선 환경에 적합하게 수정한 ATP(A reliable Transmission Protocol)와 TCP를 다중경로를 통해 통신을 할 수 있는 TCP-M 방식을 기반으로 ATP-M을 제안하였다. 성능이 더욱 개선되고 다중경로를 지원하는 ATP-M 프로토콜이 기존의 TCP, TCP-M, ATP보다 처리량이 우수함을 NS-2 시물레이션을 통해 보인다. 또한 다중경로 ATP-M에서 다중경로 개수에 따른 성능평가를 통하여 적절한 다중경로 개수를 찾는다.

**주요어** : TCP, ATP, 다중경로 ATP-M, TCP-M, 속도 기반, 전달계층

2009년 6월 30일 접수, 2009년 9월 8일 채택

<sup>1)</sup> 인제대학교 대학원 전자정보통신공학과

<sup>2)</sup> 인제대학교 공과대학 정보통신공학과

주 저 자 : 이학주

교신저자 : 장재신

E-mail; icjoseph@inje.ac.kr

## 1. 서 론

사회가 급격히 발전하고 통신을 언제 어디서나 할 수 있는 환경이 필요해짐에 따라 무선통신의 수요가 기하급수적으로 늘어나게 되면서 센서 네트워크, Ad-hoc 네트워크, 모바일 네트워크 등과 같은 다양한 형태의 무선 통신망이 출현하였다. 한편, 이러한 무선 네트워크에서는 기존 TCP(Transmission Control Protocol) 프로토콜을 사용할 경우 문제점이 발생한다. 무선 네트워크의 특징들 중에는 노드들의 이동성으로 인해 네트워크 구조가 자주 변하게 되고, 이로 인해 무선 네트워크에서 기존 TCP를 사용할 때에 성능을 떨어뜨리게 된다. 무선 통신의 제한된 대역폭과 노드의 이동성으로 인한 문제점이 유선에서 가능한 서비스를 수용하기 어렵게 하고, 이동하면서 무선으로 통신이 가능해야 유선에서의 서비스가 무선에서도 가능해진다. 유선 네트워크 구축이 어렵거나 재난 발생 지역, 전시 상황 같은 특수한 경우에는 무선으로만 이루어진 Ad-hoc 네트워크 구축이 필요하게 된다. 유선 네트워크에 적합한 TCP 프로토콜들이 있듯이 Ad-hoc 네트워크에 적합한 무선전달계층 프로토콜이 있어야 한다.

무선 전달계층프로토콜에는 I-TCP(Indirect TCP), SN OOP TCP, TCP Santa cruz, Freeze TCP, TCP-F(TCP-Feedback), Fixed RTO(Retransmission TimeOut), ATCP(Ad hoc TCP), ATP(A reliable Transport Protocol), TCP-M(TCP Multiflow) 등이 있다. I-TCP(Barke 등, 1995)는 FN(Fixed Network)와 BS(Base Station), BS(Base Station)와 MH(Mobile Host)로 두 부분으로 연결을 나누었으며, 유선 부분과 무선 부분을 혼잡 제어와 흐름 제어를 분리하여 각 연결의 특성에 맞게 한 프로토콜이다. 하지만 단대단 통신의 의미가 없어졌으며, 핸드오프가 발생할 때는 오버헤드가 커지는 단점이 있다. SNOOP TCP(Balakrishnan 등, 1995)도 무선 부분과 유선 부분을 나누었으며, BS에 패킷이 전송되어 오면 패킷을 SNOOP 캐쉬에 저장하고 전송한다. 들어온 패킷의 시퀀스번호에 따라 이미 SNOOP 캐쉬에 저장되어 있는지 없는지에 따라 동작이 달라지는 프로토콜이다. SNOOP TCP도 I-TCP와 마찬가지로 단대단 통신의 의미가 없어진 단점이 있다. Freeze TCP(Goff 등, 2000)는 단대단 통신을 유지하며, 이동성으로 인한 경로의 연결 끊김이 임박한 MH가 FH에게 ZWP(Zero Window Probe) 패킷을 전송한다. 이 패킷을 수신한 FH는 혼잡윈도우 크기를 0으로 하여 전송을 중지한 뒤에 두 단말이 다시 연결 설정이 되면 MH가 혼잡윈도우 크기를 알려 주고, 패킷 전송이 시작된다. 이 패킷을 잃어버리면 전송

을 시작할 수 없고, 무한정 기다려야하는 문제가 발생한다. 이 문제를 해결하기 위한 TCP 타이머가 Persist 타이머이며, 이 타이머는 혼잡윈도우 크기가 0인 패킷을 수신한 이후 시작되며, 일정 시간 간격으로 1바이트의 패킷을 전송하며 이에 대한 ack을 통해 혼잡윈도우 크기를 갱신한다. Persist 타이머에 의해 재연결되더라도 일정 시간을 기다려야 하는데 이를 보완하기 위한 방법(Caceres 등, 1994)도 있다. 이 방법은 재연결 되었을 때 일정 시간을 기다리지 않고 MH가 3개의 중복 ack을 전송하여 이를 FH가 수신하여 혼잡윈도우를 갱신하는 방법이다. 이동성에 의해 경로의 연결 끊김이 자주 발생하는 무선 네트워크 환경에서 경로의 연결 끊김이 임박했을 때 적절한 시간에 ZWP 패킷을 전송하기 어려운 단점이 있다. TCP-F(Chandran 등, 2001)는 경로의 연결이 끊겼을 때 송신부에 RFN(Route Failure Notification) 패킷을 전송하고, 이 RFN 패킷을 수신한 송신부는 패킷의 전송과 재전송 타이머와 혼잡윈도우 크기의 변화를 중단하고, RFN 타이머를 구동한다. 두 단말이 재연결되었을 때 송신부에 RRN(Route Reestablishment Notification) 패킷을 전송하고 이 RRN 패킷을 수신한 송신부는 멈춘 타이머의 타임아웃 값과 혼잡윈도우 크기를 기반으로 다시 전송을 재개한다. RFN 타이머가 타임아웃 되면 RRN 패킷을 수신했을 때와 같은 동작을 한다. 이는 무선 네트워크 환경에서 경로의 연결 끊김은 대부분 금방 복구되는 연결 끊김이기 때문에 RRN 패킷의 손실이나 딜레이는 재연결 되었음에도 이를 송신부가 알지 못해 무한정 기다리는 문제를 해결하기 위함이다. Fixed RTO(Dyer 등, 2001)는 재전송 타임아웃(RTO) 값을 고정시키고, 2번의 타임아웃으로 경로의 연결 끊김을 결정하는 프로토콜이다. 무선 네트워크 환경에서 2번의 타임아웃이 발생할 정도면 경로의 연결이 끊어진 경우가 대부분이다. 타임아웃 값은 타임아웃이 발생할 때 마다 타임아웃 값이 증가하는데 이 또한 무선에서 성능을 감소시키는 요인이므로 타임아웃값을 고정시키는 게 이를 해결하기 위한 방법이다. ATCP(Parsa 등, 1999)는 TCP 아래 위치하는 프로토콜이며 경로의 연결 끊김 문제뿐만 아니라 혼잡에 의한 패킷 손실과 연결 끊김, 무선랜덤오류에 의한 패킷 손실을 구분한다. 경로의 연결 끊김이나 무선랜덤오류에 의해 패킷이 손실되면 수신측에서 중복 ack 3개를 전송하게 되는데 이는 혼잡에 의한 패킷 손실이 아님에도 불구하고 송신측이 중복 ack 3개를 수신하면 혼잡 제어 기법을 수행하게 된다. 필요하지 않은 혼잡 제어 기법이 무선 네트워크 환경에서 TCP의 성능을 떨어뜨리는데 ATCP는 3번째 중복 ack을 수신하면

이를 TCP로 올려 보내지 않고, TCP의 상태를 혼잡원도우 크기가 0인 상태인 Persist 상태로 만든다. 그 뒤 ATCP가 손실된 패킷을 재전송하고, 재전송된 패킷에 대한 ack을 수신하면 이 ack을 TCP로 올려 보내고 TCP의 상태를 원상복구 시킨다. 또 혼잡에 의한 패킷 손실일 경우에는 ack 패킷과 데이터 패킷의 ECN(Explicit Congestion Notification) 플래그를 설정하고, 이 패킷을 수신한 ATCP는 아무 동작도 하지 않으므로써 TCP의 혼잡 제어 기법이 동작하도록 한다. 또 ICMP(Internet Control Message Protocol) 패킷에 의해 DU(Destination Unreachable) 패킷을 ATCP가 수신하면 TCP를 Persist 상태로 만들고 주기적으로 프로브 패킷을 전송한다. 재연결시 수신측은 중복 ack 또는 프로브 패킷에 대한 ack을 전송하고 이를 수신측의 ATCP가 수신하면 TCP의 상태를 복구하고 Slow-Start 기법에 의해 전송을 다시 시작한다. TCP santa cruz (Liu 등, 2001)는 2개의 패킷으로 딜레이를 측정하고 패킷간 딜레이로 경로의 혼잡을 발견한다. 패킷들의 구간마다 이 딜레이의 합을 구하고 딜레이의 합과 평균 패킷간 딜레이로 경로의 전송 속도를 구한다. 이를 위해 초기 혼잡원도우 값을 2로 두고 전송 속도에 기반해 패킷을 전송한다. ATP(Sundaresan 등, 2003)는 속도에 기반을 둔 전달계층 프로토콜이고, 이를 위해 패킷의 헤더에 필드 D를 추가한다. 중간 노드들은 지나가는 패킷의 Qt(Queuing Delay)와 Tt(Transmission Delay) 값을 구하고 평균 QT와 Tt값으로 갱신해둔다. 싱크 패킷에 의한 ack 패킷이나 주기적인 SACK 패킷은 각 중간 노드의 평균 Qt와 Tt의 합이 가장큰 값을 필드 D에 넣어 송신부에 전송한다. 이를 수신한 송신부는 필드 D값을 이용하여 전송 속도를 결정하고, 결정된 속도를 기반해 패킷을 전송한다. TCP-M (Thanthy 등, 2006)은 TCP를 다중경로를 사용하여 패킷을 전송하는 프로토콜이다. 단일 경로일 경우에는 연결이 끊어지면 재연결될 때 까지의 시간동안 패킷의 전송을 할 수 없는 단점이 있지만 다중 경로일 경우에는 하나의 경로가 끊어지더라도 그 경로가 재연결될 때까지 기다리지 않고 다른 경로를 통해 전송을 계속 할 수 있으므로 단일 경로에 비해 성능이 좋아짐을 알 수 있다.

무선네트워크를 위한 계층간 조합방법을 사용하는 전략적인 전송구조를 제안한 방법(안병구 등, 2007)도 있다. 또 크로스 레이어 프로토콜을 이용하여 데이터를 전송하는 방법(손진희 등, 2007)도 연구되어있다.

이제까지 제안된 무선 전달계층 프로토콜은 무선 네트워크 환경의 특징에 의해 혼잡에 의한 패킷 손실과 경로의 연결 끊김, 무선랜덤오류에 의한 패킷 손실을 구별하

여 혼잡 제어의 사용 유무를 정함으로써 무선 네트워크 환경에서 TCP의 문제점을 보완해 왔다. 무선 네트워크 환경에서 TCP 사용의 문제점은 이 외에도 연결 끊김 이후 새로운 경로의 연결이 재설정 될 때까지 전송을 할 수 없는 문제점을 가지고 있다. 이 문제를 해결 한다면 무선 네트워크 환경에서 더 좋은 성능을 가지는 프로토콜이라 할 수 있다. 경로의 연결이 끊어지더라도 여분의 경로가 있다면, 끊어진 경로의 연결이 재설정될 때까지 사용 할 수 있을 것이다. 이러한 문제점을 해결하기 위해 앞서 설명한 TCP-M(Thanthy 등, 2006)처럼 다중경로를 통해서 통신하는 방법과 혼잡과 무선랜덤오류로 인한 패킷 손실에 적절히 대응하는 ATP의 방식을 연구하여 Ad-hoc과 같은 무선 통신 환경에서 성능을 더욱 향상 시킬 수 있는 프로토콜을 제안한다. 본 연구의 2장에서는 본 연구에서 제안하고 있는 다중경로를 지원하는 ATP-M 프로토콜을 기술하며 3장에서는 다중경로를 지원하는 ATP-M의 성능분석 결과를 보이고, 4장에서는 본 연구의 특징과 장점을 요약한다.

## 2. 제안된 프로토콜 : ATP-M

TCP는 유선 전달계층 프로토콜로서 오류가 적은 유선 네트워크에서 사용되는 프로토콜이다. 패킷 손실이 적은 유선 네트워크에 최적화된 TCP를 경로의 연결 끊김, 무선랜덤오류, 높은 지연시간과 지연시간의 변화, 제한된 대역폭과 대역폭의 변화 같은 특성을 가지는 무선 네트워크에 사용하기에는 원하는 성능을 얻을 수 없다. 유선 네트워크에서 이용되는 서비스를 무선 네트워크에서 이용하기 위해서는 무선 네트워크에 적절한 전달계층 프로토콜이 절실히 필요한 실정이다. 앞장에서 설명한 여러 가지의 무선 전달계층 프로토콜이 있지만, 특정한 환경에 특화 되어 있거나, 좋은 성능을 내지 못하고 있다.

본 논문에서는 다중경로를 지원하는 ATP-M 프로토콜을 제안한다. ATP-M 프로토콜은 윈도우에 기반을 둔 전송 방법에서 속도에 기반을 둔 전송 방법으로 수정하였으며, Slow-Start 기법을 Quick-Start 기법으로 수정하여 한번의 RTT로 경로의 이용 가능한 대역폭을 찾고 그 대역폭을 전송속도로 환산하였다. 그리고 단일 경로에서 성능 향상 폭은 제한되어 있기 때문에 다중경로를 지원 할 수 있도록 AOMDV(Marina 등, 2001) 라우팅 프로토콜을 ATP-M에 적용하였다.

### 2.1 ATP 프로토콜 알고리즘

앞에서 언급했듯이 무선 네트워크의 특성에는 노드들은 이동성을 가지며, 제한된 대역폭과 대역폭의 변화, 높은 지연시간과 지연시간의 변화, 무선랜덤오류 등이 있으며 경로의 연결이 자주 단절된다. TCP를 무선 네트워크 환경에서 사용할 때 전송률이 떨어지는데 무선 네트워크 환경에서도 전송률을 증가시킬 수 있는 프로토콜이 필요하다. 앞서 설명한 ATP가 무선 네트워크 환경 특성에 TCP의 기법들을 적절하게 수정한 프로토콜이다.

ATP는 속도에 기반을 둔 전송 방법이며 TCP의 윈도우에 기반을 둔 전송 방법의 문제점을 해결하고, 무선 통신망의 변화하고 제한된 대역폭에 알맞은 전송 속도를 설정할 수 있다. 혼잡 제어(Congestion Control) 기법과 재전송(Reliability) 기법을 분리하여 혼잡 제어 기법은 네트워크의 피드백을 이용하여 수행하고, 재전송 기법은 SACK 패킷과 피드백의 수신으로 수행한다. 혼잡 제어를 쉽게 하기 위해 네트워크에서 중간 노드는 혼잡 정보를 이용할 수 있는 속도로 환산하여 제공하고, 데이터 패킷에 피드백하여 환산된 정보를 알려준다. 재전송은 수신부에서 SACK 패킷을 RTT가 되기 전에 대역폭에 여유가 있을 때 전송하고, 이때의 SACK 패킷 TCP와는 다르게 복합적인 ack 패킷이고 ATP는 SACK 패킷의 정보에 의지하여 재전송한다.

ATP는 송신부, 수신부, 중간 노드에 따라 각각의 역할이 있으며, 그 역할에 대해서 설명한다. 중간 노드에서는 경로의 이용 가능한 속도를 환산하여 그 정보를 ack 패킷에 피드백한다. 이용 가능한 속도의 계산은 패킷의 평균 큐잉(queueing) 지연 시간과 평균 전송 지연 시간을 이용하고, 이 값들은 패킷이 지나 갈 때마다 계산한다. 수신부에서는 전송속도를 피드백하고, 이로 인해 송신부에서 순간적으로 많은 양의 데이터를 전송하지 않게 한다. 또 패킷 손실의 정보를 주기위해 피드백(SACK)을 사용하고 모든 패킷에 대해 피드백을 주지 않고 주기적으로 피드백을 준다. 송신부에서는 TCP의 Slow-Start를 사용하지 않고 Quick-Start를 사용하여 수신부에서 얻는 피드백 정보를 토대로 전송 속도를 결정한다. 혼잡 제어는 수신부로부터 피드백된 속도가 현재 속도보다 높으면 전송속도를 증가시키는 증가 단계가 되며, 현재 속도보다 낮으면 속도를 감소시키는 감소 단계가 된다. 이외에는 전송속도를 지속 시킨다.

하지만 ATP도 노드 이동에 의해 발생하는 잦은 경로 끊김에는 적절하게 대처하지 못하고 있다. 이러한 부분을 보완하기 위해 TCP-M 프로토콜(Thanthry 등, 2006)의

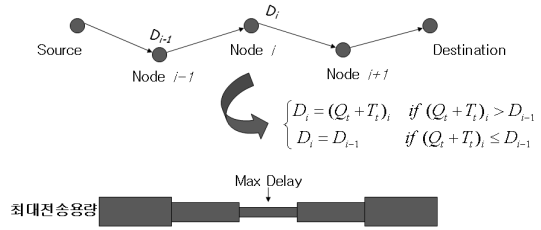


그림 1. 중간 노드

다중 경로 기법을 ATP 프로토콜에 접목하여, 새로운 프로토콜을 제안 한다. 본 연구에서는 이 새로운 프로토콜을 시뮬레이션 할 수 있도록 NS-2 네트워크 시뮬레이터를 사용하여 구현하였다. NS-2는 현재 널리 사용되고 있는 네트워크 시뮬레이터로 네트워크를 연구하기 위하여 개발된 이벤트 기반의 시뮬레이터이다. 이를 통해 다양한 유선 네트워크(예 : 전달계층 프로토콜, 사용자 데이터그램 프로토콜, 파일 전송 규약, 아이피 텍스트 전송 규약)와 무선 네트워크(예 : Ad-hoc, 무선랜, 모바일 인터넷 프로토콜) 등을 시뮬레이션 할 수 있다.

### 2.2 ATP 프로토콜 구현

TCP가 가지고 있는 기능 중에는 윈도우에 기반을 둔 전송 제어, Slow-Start, 패킷 손실에 기반을 둔 혼잡 제어, 혼잡 발생시 윈도우크기를 반으로 감소, 재전송 타임아웃 등이 있다. TCP를 무선 네트워크에 곧바로 적용할 경우에는 다양한 문제가 발생한다. 무선 네트워크에서 TCP 프로토콜을 사용할 경우 문제가 되는 TCP의 기능을 ATP 프로토콜은 다음과 같이 수정하였다. 윈도우 기반의 전송 제어를 속도에 기반을 둔 전송기법으로 수정하며, Slow-Start 기법 대신에 퀵-스타트 기법, 패킷 손실에 기반을 둔 혼잡 제어에서 네트워크의 정보를 피드백을 통해 얻으며 이를 혼잡 검출과 제어에 사용하는 기법, 재전송 타임아웃을 이용하여 재전송을 하지 않고 이용 가능한 대역폭 정보를 가지는 피드백(SACK)을 통해 재전송을 한다.

그림 2는 송신부의 데이터 흐름도이고 그림 3은 송신부의 SACK 패킷 수신시의 동작 pseudo-code이다.

TP 연결 설정 과정을 통해 네트워크의 정보를 피드백(SACK)을 통해 알게 되었을 때 패킷 전송을 시작하고, 처음 패킷이 전송된 후에는 RateSndTimer에 의해 패킷을 전송한다.

경로의 연결 설정을 위한 SYN 패킷에 의한 SACK 패킷을 수신시 데이터 패킷을 전송하게 되고, 데이터 패킷 전송과 동시에 RateSndTimer가 구동된다. 그림 4와 같이 SYN 패킷에 의한 SACK 패킷을 수신하게 되면, 피드백

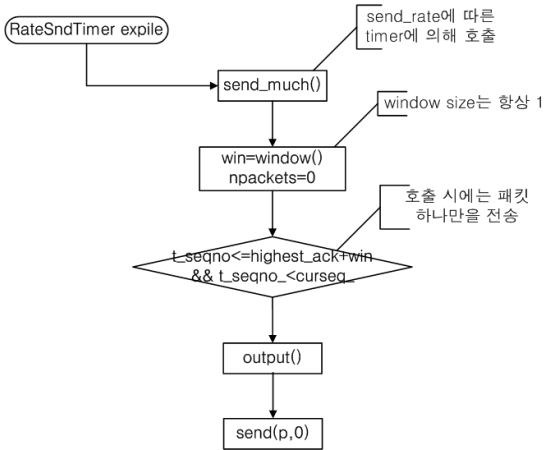


그림 2. 송신부 송신

```
// 패킷 수신
// 수신한 패킷의 종류
if(data == SYN packet & SACK packet){
    // SYN packet에 의한 SACK packet일 때
    R = 1/Avg(D);
    S = R;

    // 타이머 구동 //
    packet_trans_timer_.sched(0);
}
else if(data == SACK packet){
    // SACK packet일 때
    R = 1/Avg(D);
    if(S < R - 파이 * S) S = S + (R-S)/k;
    else if(S > R) S = R;
    else S = S;
}
// 송신측 s값 업데이트
```

그림 3. SACK 패킷 수신시 pseudo-code

된 전송속도에 의해 데이터 패킷을 전송하고, 패킷을 전송하고 있는 동안 수신부로부터 피드백(SACK)을 수신하게 되면, 현재의 전송속도와 피드백된 전송속도를 비교하여 증가, 감소, 지속의 3가지 단계를 통해 전송속도를 변경하고, RateSndTimer의 타임아웃값을 결정한다.

네트워크 정보를 얻기 위해서 중간 노드들은 노드의  $Q_t$ ,  $T_t$ 값을 구하고, SACK 패킷이 송신부로 전송될 때 피드백 된다. 그림 5는 중간 노드에서  $Q_t$ 값과  $T_t$ 값을 구하는 pseudo-code이다.

$Q_t$ 는 큐잉지연시간이며,  $T_t$ 는 전송지연시간이다. 각 중간노드마다 평균  $Q_t$ 와  $T_t$ 값을 계산하고, SACK 패킷은 중간노드들 중 가장 큰 평균  $Q_t+T_t$ 값을 피드백하여 송신부에 전송된다.

경로의 연결설정을 위해 전송되어온 SYN 패킷을 수신하면 수신부 노드의  $Q_t+T_t$  값을 SACK 패킷에 피드백하여 전송하고, 데이터 패킷을 수신했을 때는  $Q_t$ 와  $T_t$ 값을

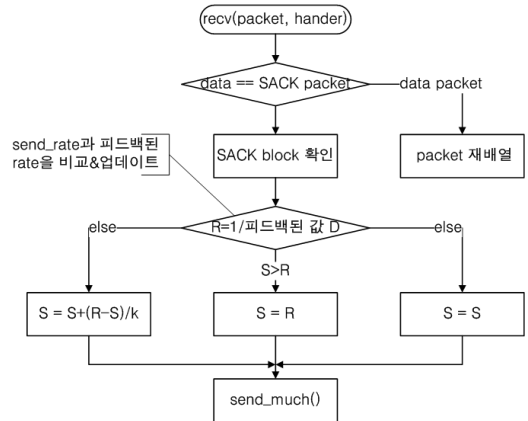


그림 4. 송신부 송신

```
if(data == SACK packet){
    if(Avg(Qt) + Avg(Tt) > field D)
        field D = Avg(Qt) + Avg(Tt);
}
else if(data == data packet){
    Current(Qt) = queuing 지연시간 계산;
    Current(Tt) = link-of-head 시간 계산;

    // 중간노드의 field 값에 저장
    Avg(Qt) = 알파 * Avg(Qt) + (1-알파) * Current(Qt);
    Avg(Tt) = 알파 * Avg(Tt) + (1-알파) * Current(Tt);

    if(Avg(Qt) + Avg(Tt) > field D)
        field D = Avg(Qt) + Avg(Tt);
}
}
```

그림 5.  $Q_t$ 값과  $T_t$ 값을 구하는 pseudo-code

```
if(data == SYN packet){
    // TCP 연결 설정 중 수신 한 SYN packet
    Avg(D) = Current(D);

    // 타이머 구동 //
    SACK_timer_.sched(0);
}
else if(data == data packet){
    Avg(D) = 베타 * Avg(D) + (1-베타) * Current(D);
}
}
```

그림 6. 수신부 pseudo-code

측정하고 평균  $Q_t$ ,  $T_t$ 값에 의해 평균  $D$ (평균  $Q_t+T_t$ )값을 업데이트 해 놓고, SACK 패킷 전송시 이  $D$ 값을 피드백하여 전송한다.

### 2.3 ATP-M 프로토콜 제안 및 구현

전송속도에 기반을 둔 ATP 프로토콜은 무선 네트워크 환경에서 좋은 성능을 보이지만, 노드의 잦은 이동으로 인한 경로의 연결 끊김의 영향으로 최상의 성능을 내지는 못하고 있다. ATP는 잦은 경로 단절 환경에서 더 좋은 성능을 낼 수 있는 여지를 가지고 있으며, 다중경로를 지원한다면 ATP는 최상의 성능을 낼 수 있다. 다중경로 환경을

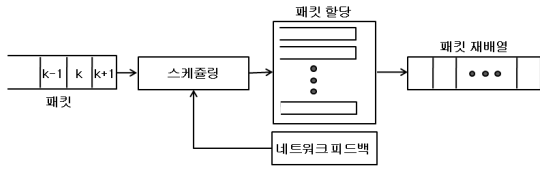


그림 7. ATP-M의 전체 흐름도

```
// 라우팅 정보에 접근
aomdv_rt_entry *rt = rtable.rt_lookup(ih->daddr());
Packet *p;

for(int i=0; i<10; i++)
    if(rt->link == ltarget[i])
        output(seqno, reason, rt);

// 라우팅 정보에 접근
```

그림 8. 라우팅 테이블 접근

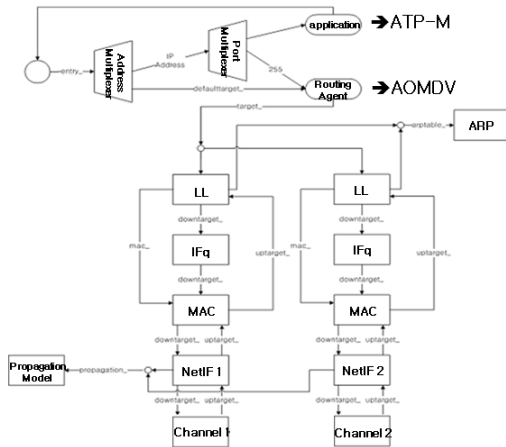


그림 9. 다중경로를 지원하는 모바일 노드 구조

지원하는 TCP-M(Thanthry 등, 2006)의 기법을 ATP 프로토콜에 접목시키기 위해 다중경로를 지원하는 AOMDV 라우팅 프로토콜을 이용하였다. AOMDV 라우팅 프로토콜은 1개 이상의 경로를 얻을 수 있으며, ATP를 확장한 후에 추가하였다. 이를 ATP-M이라 한다. ATP-M 프로토콜은 한 개 이상의 경로를 유지하면서 경로의 연결이 끊기더라도 다른 경로를 통해 전송을 계속 할 수 있도록 하였다.

ATP-M은 그림 7과 같이 송신측에서 전송 할 패킷을 각 경로에 할당 하는데 각 경로에 알맞는 패킷을 할당하기 위해 네트워크의 피드백을 이용한다. 네트워크의 피드백은 송신부로부터 주기적으로 전송되는 SACK 패킷에 피드백되고, 송신부는 이 정보를 전송속도로 환산한다. 수신부에서는 송신부로부터 각 경로를 통해 전송되어온 패킷들을 재배열한다.

ATP-M을 구현하기 위해 NS-2 2.32에 구현 되어 있는 tcp.{cc, h}, mac-802\_11.{cc, h}, drop-tail.{cc, h} 코드를 수정하였으며, aomdv.{cc, h}, aomdv\_packet.h, aomdv\_logs.cc, aomdv\_rqueue.{cc, h} 파일을 추가하여 제안한 ATP-M이 다중경로를 이용할 수 있도록 포팅하였다.

중간노드는 각 노드의 최대 지연시간을 계산하며, 최대 지연시간은 큐잉 지연시간(Qt)과 전송지연시간(Tt)의 합이다. 이 계산을 위해 mac-802\_11.cc 파일의 mac802\_11::transmit(Packet \*p, double timeout) 함수에서 패킷이 노드에 들어오는 시점의 시간을 저장하고, drop-tail.cc 파일의 DropTail::enqueue() 함수에서 패킷이 노드의 큐에 들어오는 시간을 저장한다. 이렇게 저장된 시간을 mac-802\_11.cc 파일의 mac802\_11::recv(Packet \*p, Handler \*h) 함수에서 현재 노드를 지나가는 패킷의 큐잉지연시간과 전송지연시간을 계산하고, 노드의 평균 큐잉지연시간과 평균 전송지연시간의 합을 계산한다. 이 값을 D값이라 하고, 네트워크의 피드백 정보로 송신부에 피드백된다.

또 ATP-M이 다중경로를 지원하기 위해 AOMDV 라우팅 프로토콜을 포팅하였으며, 기존의 전달계층 프로토콜은 라우팅 정보를 알 수 없었으나, ATP-M은 라우팅 정보를 알 수 있으며, 이 정보를 이용하여 여러 경로를 이용할 수 있다.

tcp.cc 파일의 TcpAgent::send\_much() 함수에서 라우팅 테이블의 정보를 이용하여 각 경로에 알맞은 양의 패킷을 할당한다.

ATP-M의 모바일 노드가 다중경로를 지원하기 위해 LL(Link Layer), IFq(Interface queue), MAC, NetIF(Network InterFace), 채널(Channel)을 이용 할 수 있는 경로의 수와 같이 만들어 준다. 전송 할 패킷이 Address Multiplexer까지 도달하면 패킷의 목적지를 확인하고 아래 단계로 보내준다. Routing Agent에서 경로를 확인하고 패킷을 전송한다. 그림 9는 다중경로를 지원하는 모바일 노드 구조이다.

다중경로를 지원하는 ATP-M은 다음과 같은 알고리즘을 가진다. ATP를 그림 10과 같이 두 개의 계층으로 나누어 상위 전송계층은 상위계층과 통신, 경로의 연결설정, 패킷 재배열, 패킷 스케줄링의 역할을 담당하고, 하위 계층은 데이터 전송 역할을 한다. 데이터 패킷은 본래의 시퀀스번호와 identifier 필드를 이용하여 전송하고, 이 필드를 수신부에서 패킷 재배열을 하는데 사용한다. 다중경로 연결에 의한 패킷임을 나타내기 위해 기존 TCP의 reserved bit를 이용한다. 그림 11은 다중경로를 지원하는 ATP-M의 헤더 포맷이다.

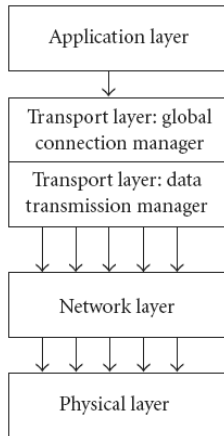


그림 10. 2개의 전송계층

Source port		Destination port	
Sequence number			
Acknowledgment number			
HLEN	Reserved	Code bits	Window
Checksum		Urgent pointer	
Global connection identifier			
Global sequence number			
Data			
...			

`[SPLT] URG ACK PSH RST SYN FIN`

그림 11. 다중경로 ATP 헤더 포맷

그림 11은 TCP 포맷에서 다중경로를 지원하기 위해 기존의 시퀀스 번호와 ack 번호 외에 global connection identifier와 global sequence number 필드를 가지고 있다. 또 다중 경로를 통하는 패킷임을 code bits의 SPLT 필드를 통해 설정한다.

### 3. 성능평가

앞장에서 제안한 다중경로를 지원하는 ATP-M의 성능을 평가하기 위해 우선 네트워크에 최적화된 TCP, 그리고 TCP-M, ATP와의 성능을 비교하였다. 먼저 TCP, TCP-M, ATP 그리고 다중경로를 지원하는 ATP-M의 네 가지 프로토콜의 성능을 비교하기 위해 패킷 손실률에 따른 처리량으로 성능평가를 하였고, 다중경로를 지원하는 ATP-M 프로토콜의 가장 적절한 다중경로 연결 수를 찾기 위해 다중경로 수를 변화 시키면서 성능평가를 하였다.

제안한 ATP-M이 기존의 ATP보다 약 90% 성능이 증가하였으며, 적절한 다중경로의 개수 일 때는 그 성능 향상이 약 170%까지 나타남을 확인하였다. 또 TCP-M에

비해서도 약 5% 성능향상이 되었다. 적절한 다중경로 수는 네트워크 환경의 노드 수에 따라 달라졌으나 3개의 경로일 때 그 성능이 가장 우수했다.

#### 3.1 성능평가 환경

제안한 프로토콜의 성능평가를 위하여 NS-2 네트워크 시뮬레이터를 사용하여 각 프로토콜을 구현한 후 평가하였다. NS-2에 구현되어 있는 TCP 코드를 기반으로 ATP를 구현하였으며, 다중경로를 지원하는 ATP-M을 구현하기 위하여 AOMDV 라우팅 프로토콜을 사용하고, ATP를 확장 구현하였다. ATP-M은 변화하는 네트워크 환경에 잘 대응할 수 있도록 네트워크 피드백 정보를 이용하는 프로토콜이므로 노드의 이동성으로 네트워크상태가 자주 변화하거나 혼잡이 발생하는 환경을 기본으로 하였다.

무선 네트워크 환경에서 TCP, ATP, 다중경로를 지원하는 TCP-M, ATP-M의 성능평가를 하기 위한 시뮬레이션 환경은 다음과 같다.

성능평가 환경은 20개의 노드가 이동성 모델이 random way-point 인 환경이다. random way-point는 노드들이 초기 위치에서 시작해 방향, 속도, 시간을 정해 준 후 이동하며, 노드들 마다 방향과 속도는 랜덤값을 가진다. 방향은 랜덤으로 지정되나 경계선과 마주치면 그 대칭된 각과 거리만큼 움직이며 속도는 처음 속도가 끝까지 유지된다. 그러나 노드들마다 같은 속도를 가지지 않으며, 처음 이동한 노드는 도착점에 도달하면 잠시 멈추고, 이 시간 동안 다음 방향, 속도 그리고 시간을 랜덤으로 정한 후 다음 목적지로 이동한다.

트래픽은 FTP를 사용하며 NS-2에 있는 FTP 트래픽 생성기를 사용하였다. FTP 트래픽 생성기는 ATP-M에 연결된다.

성능평가 척도로는 시뮬레이션 시간동안에 성공적으로 전송된 패킷의 수이다. 위와 같은 시뮬레이션 환경을 이용하여 제안한 다중경로를 지원하는 ATP-M의 시뮬레이션을 수행하여 성능평가 척도에 따라 결과를 분석하였다.

#### 3.2 성능평가 결과

앞장에서 언급한 시뮬레이션 환경을 바탕으로 각각의 프로토콜의 시뮬레이션 결과는 다음과 같다. 먼저 TCP, ATP, 다중경로를 지원하는 ATP-M의 성능평가 척도로 처리량에 대한 결과 그래프이다. 본 결과값은 패킷 손실율을 점차 증가시켜 가면서 해당 성능평가를 한 결과이다.

그림 12의 결과 그래프를 보면 무선 네트워크 환경에서 ATP는 TCP보다 좋은 성능을 보이고 있고, 다중경로

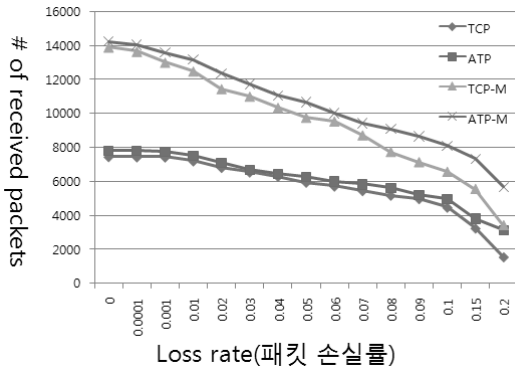


그림 12. TCP, ATP, TCP-M, 다중경로를 지원하는 ATP-M

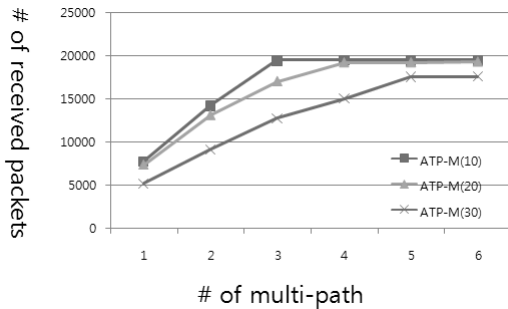


그림 13. 경로 수에 따른 ATP-M

를 지원하는 ATP-M이 TCP와 ATP에 비해 더 좋은 성능 결과를 보이고 있음을 확인 할 수 있다. 하나의 경로만을 이용하는 ATP-M은 ATP와 같은 결과를 보이지만, 위 그림 12의 결과에서는 두 개의 다중경로를 가지는 ATP-M이며, 그로 인해 기존의 ATP보다 높은 성능을 보인다. 또 TCP-M 보다 ATP-M의 성능이 좋게 나왔는데, 이는 ATP의 기법들이 다중경로에서도 잘 적용 되었다는 것을 알 수 있다.

다중경로의 연결수가 늘어남에 따라 전송 처리량이 증가하다가 3개의 다중경로 연결을 가지는 지점부터 증가하는 처리량이 줄어드는 것을 결과가 나왔다. 이는 경로의 수가 많을수록 전송 할 수 있는 데이터 패킷이 늘어나 처리량이 늘어났지만, 다중경로의 수가 항상 원하는 수만큼 가지지 있지 않고, 모든 경로가 다 좋은 망 상태가 아니기 때문에, 지연시간이 긴 경로의 패킷으로 인해 수신부에서는 패킷 재배열의 오버헤드가 발생한다. 이것이 세 개의 경로수 까지는 처리량이 증가하지만 그 수를 넘어서면서 부터 처리량 크게 증가하지 않는 이유이다.

## 4. 결 론

본 연구에서는 무선 네트워크에 알맞은 전달계층 프로토콜을 제안하기 위해 기존의 TCP를 무선 네트워크에 적절하게 수정한 ATP 프로토콜을 NS-2를 사용하여 직접 구현하였고, 보다 좋은 성능을 얻기 위해 ATP를 다중경로를 지원 할 수 있도록 확장 구현하였다. ATP를 다중경로를 지원하도록 확장한 ATP-M이 본 논문에서 제안하는 전달계층 프로토콜이다. 다중경로를 지원하기위해 라우팅 프로토콜은 AOMDV를 사용하였고, AOMDV 라우팅 프로토콜을 ATP-M에 포팅하였다. AOMDV를 이용하여 다중경로의 연결 수를 결정하고 이용하였다.

제안한 ATP-M이 기존의 ATP보다 약 90% 성능이 증가하였으며, 적절한 다중경로의 개수 일 때는 그 성능 향상이 약 170%까지 나타남을 확인하였다. 또 TCP-M에 비해서도 약 5% 성능향상이 되었다. 적절한 다중경로 수는 네트워크 환경의 노드 수에 따라 달라졌으나 3개의 경로일 때 그 성능이 가장 우수했다. 본 연구에서는 Ad-hoc 무선 네트워크를 기본으로 하여 적절한 프로토콜을 제안하였고, 차후 연구 방향으로는 여러 무선 네트워크가 혼합되어 있는 환경에서도 좋은 성능을 충족시킬 수 있는 프로토콜을 연구해야 할 것이며, 그 연구들은 통합 무선 네트워크 환경에서도 효과적으로 데이터를 전송 할 수 있는 것이다.

## 참 고 문 헌

1. A. Barke abd B. R. Badrinath, (1995), "I-TCP: Indirect TCP for mobile hosts." in Proc. 15th Int. Conf. Distributed Bomputing Systems, Vancouver, BC, Canada, pp. 136-143.
2. C. Parsa and J.J. Garcia-Luna-Aceves, (1999), "Improving TCP Congestion Control Over internets with Heterogeneous Transmission Media", Proc. IEEE ICNP 99.
3. H. Balakrishnan, S. Seshan, and R. H. Katz, (1995), "Improving reliable transport and handoff performance in cellular wireless networks," ACM Wireless Networks, Vol. 1, No. 4, pp. 469-482.
4. J. Liu and S. Singh., (2001), ATCP: TCP for mobile ad hoc networks. IEEE Journal on Selected Areas in Communications, 19(7):1300-1315.
5. K. Chandran et. al., (2001), "A Feedback-based scheme for improving TCP Performance in ad hoc wireless networks," IEEE Personal Comm. Magazine.



6. K. Sundaresan, V. Anantharaman, H.-Y. Hsieh, and R. Sivakumar, (2003), "ATP: A Reliable Transport Protocol for Ad-hoc Networks," Proc. ACM MobiHoc.
7. M. K. Marina and S. R. Das, (2001), "On-demand multipath distance vector routing in ad hoc networks," in Proceedings of the International Conference for Network Protocols (ICNP '01), pp. 14--23, Riverside, Calif, USA.
8. N. Thanthy, et.al. (2006), TCP-M: Multiflow Transmission Control Protocol for Ad Hoc Networks, EURASIP Journal on Wireless Communications and Networking, Article ID 95149, 16 pages.
9. R. Caceres, L. Iftode,, (1994), "Improving the performance of Reliable Transport Protocols in Mobile Computing Environments", IEEE JSAC, Special Issue on Mobile Computing Network.
10. S. McCanne and S. Floyd, "NS network simulator," version 2.32, <http://www.isi.edu/nsnam/ns>
11. T. Goff, J. Moronski, and D.S. Phatak, (2000), "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments," INFOCOM.
12. T. D. Dyer and R. V. Boppana, October 200, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," Proceedings of ACM Mobihoc, pp. 56-66, 1.
13. 안병구, 김남수, 김도현 (2007), "A Power Saving Transmission Architecture Using Cross-Layer Combination Method for Mobile Ad-hoc Wireless Networks", 한국인터넷방송통신TVgkr회 논문지, pp. 21-29, 제7권 제1호.
14. 손진희, 이형근, 이장연, 조진웅 (2007), "무선 센서 네트워크에서 클러스터링 프로토콜의 크로스 레이어 설계에 대한 연구", JCCI 2007 한국통신학회.



**이 학 주** (maniac4444@nate.com)

2007 인제대학교 전자정보통신공학부 공학사  
2009 인제대학교 전자정보통신공학과 공학석사

관심분야 : 무선 네트워크(모델링, 시뮬레이션, 스케줄링), 무선네트워크용 전달계층 프로토콜



**장 재 신** (icjoseph@inje.ac.kr)

1990 동아대학교 전자공학과 공학사  
1992 한국과학기술원(KAIST) 전기및전자공학과 공학석사  
1998 한국과학기술원(KAIST) 전기및전자공학과 공학박사  
1997~2002 (주) 삼성전자 정보통신 총괄 책임연구원  
2008~2009 Iowa 주립대 방문연구원  
2002~현재 인제대학교 정보통신공학과 조교수

관심분야 : 이동통신 네트워크, 무선인터넷, 무선 LAN, 센서네트워크



**이 종 협** (icjhlee@inje.ac.kr)

1984 고려대학교 산업공학과 공학사  
1986 한국과학기술원(KAIST) 산업공학과 공학석사  
1996 한국과학기술원(KAIST) 산업공학과 공학박사  
1986~2004 한국전자통신연구원 책임연구원(팀장)  
2004~현재 인제대학교 정보통신공학과 부교수

관심분야 : High-speed Network Design and Routing, Switch and Router Technology, Advanced Network Protocols, Sensor and Mobile Network