

무선 멀티 홉 환경에서 ARQ를 통한 재전송 지속성 관리 기법

오 봉 환*, 김 석 규*, 이 재 용°

Retransmission Persistence Management with ARQ in Multi-Hop Wireless Network

Bong-Hwan Oh*, Seog-gyu Kim*, Jaiyong Lee°

요 약

무선 통신 기술이 점점 발전함에 따라 이제는 하나의 전송 경로 안에 다중의 무선 구간이 존재하는 무선 멀티 홉 환경이 점점 일반화 되고 있다. ARQ는 무선 구간에서 사용되는 가장 일반적인 프로토콜로서 무선 링크에서 발생하는 패킷 손실을 복구함으로써 무선 구간의 성능을 보장한다. 하지만 ARQ는 근원적으로 자신이 속한 무선 링크의 성능만을 보장하기 때문에 다중의 무선 구간이 존재할 경우에는 전체 성능을 보장하는데 문제가 생길 수 있다. 따라서 본 논문에서는 무선 멀티 홉 환경에서 전체 전송 경로의 성능을 보장하기 위한 재전송 지속성 관리 기법을 제안한다. 제안된 방법은 무선 멀티 홉 환경에서 ARQ간의 링크 정보 교환을 통해 ARQ 스스로 무선 병목링크를 인지하고 이에 따라 재전송 지속성을 조절함으로써 전체 성능을 보장한다. OPNET 시뮬레이터를 통해 제안된 방법이 전체 무선구간의 요구조건 안에서 전체 성능을 향상시키는 것을 확인 할 수 있었다.

Key Words : ARQ, multihop wireless network, retransmission persistence, E2E performance, link layer protocol

ABSTRACT

Multi-hop wireless networks has become common phenomenon according to a development of wireless communications and an increase of usage of wireless link. ARQ is one of the common protocols in link layer and can improve a link performance recovering packet loss in wireless link. However, ARQ cannot sufficient to assure a end-to-end performance because ARQ only manages the ARQ's own link. In this paper, we propose a new retransmission persistence for ARQ in multi-hop wireless network to satisfy the end-to-end performance. The proposed scheme can aware a bottleneck link according to the exchange of link information between ARQs and can support the end-to-end performance by managing a retransmission persistence. OPNET simulator is used to evaluate the performance of the proposed scheme and results show that proposed scheme can improve the end-to-end performance with satisfying a requirement of entire wireless section.

※ 본 연구는 Seoul R&BD Program(WR080951)의 지원을 받아 연구되었음.

※ 본 연구는 미래창조과학부 및 정보통신기술연구진흥센터의 정보통신·방송 연구개발사업의 일환으로 수행하였음. [1491104001, 국가 공공안전서비스를 위한 LTE기반 재난통신 시스템·단말 개발]

♦ First Author : Yonsei University, Department of Electrical and Electronic Engineering, crusader27@yonsei.ac.kr, 학생회원

° Corresponding Author : Yonsei University, Department of Electrical and Electronic Engineering, jy1@yonsei.ac.kr, 종신회원

* Andong University, Department of Information Communication Engineering, sgkion@andong.ac.kr, 종신회원

논문번호 : KICS2014-09-342, Received September 3, 2014; Revised September 23, 2014; Accepted October 13, 2014

I. 서 론

네트워크 기술의 발전과 특히 무선 단말 기술의 향상으로 인해 네트워크에서 무선 구간 부분들이 점점 증가함에 따라 이제는 하나의 전송 구간에서 하나의 무선 링크가 아닌 다중의 무선 링크들이 존재하는 무선 다중 홉 환경이 되어 가고 있다. 또한 이제는 무선 네트워크에서도 단순한 서비스를 넘어서 화상 전화, HD 스트리밍 서비스 등 더 좋은 QoS를 보장해야 하는 서비스들이 등장하고 있고 앞으로 이러한 추세는 계속 되어질 것으로 예상 된다. 하지만 아직까지 무선 네트워크는 유선 네트워크에 비해 한정된 자원을 가지고 있기 때문에 서비스의 성능을 향상시키기 위해서는 무선 네트워크의 성능을 향상 시키거나 또는 제안된 네트워크의 성능 안에서 이를 효율적으로 사용하는 것이 중요한 요소가 될 것이다.

ARQ는 무선 구간에서 신뢰성 있는 데이터 전송을 위해서 가장 폭 넓게 사용되는 link layer 프로토콜이다. ARQ가 무선 구간에서 가지는 장점 중에 하나는 ARQ기반의 error(loss) 복구 방법이 특정 무선 구간에서 손실 복구를 최적화 할 수 있기 때문에 상위 layer에서(특히 transport layer 프로토콜에서) 무선 구간의 error(loss)를 인지하지 못하게 하여 전체 성능을 향상 시키게 하는 것이다. 하지만 비록 ARQ의 무선 구간 error(loss) 복구가 상위 layer의 성능에 도움이 된다 할지라도, 모든 ARQ의 동작이 상위 layer에 도움이 되는 것은 아니다. ARQ가 안좋은 영향을 미치는 요인 중에 하나는 무선 구간 중에 발생하는 error(loss)를 복구하느라 발생하는 추가적인 delay 이다. 상위 layer는 (특히 transport layer에서) 네트워크의 성능을 BDP(bandwidth delay product)로 예측을 하게 되고 이를 위한 중요한 파라미터는 RTT(round trip time)이다. 하지만 ARQ의 추가적인 delay는 상위 layer의 정확한 RTT 측정을 방해하게 되고 이는 상위 layer로 하여금 부정확한 RTT를 예측하게 하여 성능 저하를 발생 시킬 수가 있다. 따라서 ARQ가 어떤 방식으로 동작하느냐에 따라 전체 경로의 성능에 영향을 미치게 된다.

ARQ의 복구 성능에 가장 큰 영향을 미치는 것은 ARQ의 재전송 지속성이다. ARQ의 재전송 지속성은 전송되는 데이터 frame이 손실 되었을 경우 손실된 frame를 복구하기 위해서 허용되는 재전송 횟수를 의미한다. 따라서 재전송 지속성이 높은 경우에는 반복된 frame 손실이 일어나도 충분한 시간을 가지고 복구를 수행할 수 있으나 이에 따른 복구 시간이 필요하

게 되어 전체 경로의 딜레이 성능이 나빠진다. 반대로 재전송 지속성이 낮은 경우에는 반복된 frame 손실에 취약할 수 있으나 frame 손실로 인해 발생하는 딜레이를 줄일 수 있다.

지금까지 ARQ의 재전송 지속성을 통해 전체 경로의 성능을 향상시키고자 하는 연구들이 진행되어 왔다^[1]. 이 연구들은 공통적으로 네트워크 상황에 따라 재전송 지속성을 관리하는 방법에 대해서 제안하고 있으며 이것이 재전송 지속성을 고정하는 방식보다 더 좋은 성능을 얻을 수 있다는 것을 보였다. 하지만 이러한 연구들이 ARQ의 동작을 통해서 전체 성능을 향상 시킬 수 있었던 이유 중의 하나는 전체 구간에서 무선 구간을 one-hop wireless 구간으로 가정했기 때문이다. 따라서 기존의 ARQ의 동작만으로 전체 성능을 향상 시킬 수가 있었다. 하지만 이러한 방식들은 one-hop wireless 구간에서 multi-hop wireless 구간으로 확장될 경우 문제는 달라진다. 멀티 홉 환경의 경우 다중의 무선 구간이 존재하게 되고 이 다중의 무선 구간 모두는 전체 성능에 영향을 미치게 된다. 따라서 기존의 ARQ의 동작만을 이용한 방법들은 전체 구간의 성능을 향상시키는데 충분한 방법이 되지 못한다. 결국 무선 전체 구간의 성능을 보장하기 위해서는 각각의 무선 구간에서 동작하는 ARQ들이 서로 협력적인 동작을 수행해야 만이 전체 성능을 보장할 수 있게 된다.

이와 같은 문제를 해결하기 위해서 본 논문에서는 무선 멀티 홉 환경에서 ARQ를 통해 종단간의 성능을 만족시키기 위한 협력 재전송 지속성 관리 방법을 제안하였다. 제안된 방법은 ARQ 정보 교환 방법을 통해 ARQ 간의 링크 정보를 서로 공유하고 이를 통해 각각의 ARQ들이 스스로 병목 링크를 인지하여 재전송 지속성을 조절함으로써 전체 성능을 보장한다. 또한 무선 구간의 요구조건에 따라 요구조건을 만족시키면서 성능을 향상시킨다.

본 논문의 구성은 다음과 같다. 제 II장에서는 제안된 방법에 대한 배경지식 및 연구 동기에 대해서 설명하고 제 III장에서는 제안된 방법에 대해서 설명한다. 제 IV장에서는 시뮬레이션을 통해서 제안된 방법의 성능을 평가하고 마지막으로 제 V장에서는 본 논문의 결론을 맺는다.

II. 배경지식 및 연구 동기

일반적으로 ARQ의 성능의 주된 요인은 ARQ의 재전송 지속성에 있다. ARQ의 재전송 지속성이 큰

값을 가지는 경우에는 다음과 같은 장점을 가질 수 있다. 링크의 높은 패킷 손실률에도 불구하고 충분한 재전송으로 패킷을 복구하여 무선 구간에서의 패킷 손실을 줄일 수가 있다. 이는 상위 layer 특히 transport layer의 congestion control의 감소를 방지하여 전체 성능의 향상을 가져올 수 있다. 하지만 단점으로는 복구하는데 소요되는 추가적인 delay가 발생되게 되고 이러한 현상은 단지 링크의 delay를 증가 시키는 것뿐만 아니라 전체 경로의 delay (RTT)에도 영향을 미치게 되어 상위 layer에서 네트워크를 예측하는데 어려움을 겪게 한다. ARQ의 지속성이 작은 값을 가지는 경우에는 다음과 같은 장, 단점이 존재한다. 위의 경우와는 반대로 재전송 지속성 값이 작은 값을 가지고 있기 때문에 무선 구간에서의 패킷 손실을 복구하는 성능은 부족할 수 있으나 추가적인 delay를 발생시키지 않아 상위 layer에서 delay로 인한 문제를 발생시키지 않는다.

앞 장에서 설명한 것과 같이 ARQ의 재전송 지속성을 통해 전체 경로의 성능을 향상시키고자 하는 연구들이 진행되어 왔다¹⁻⁸⁾. [1,2]의 연구에서는 무선 링크 상에서 발생하는 패킷 손실률을 측정하고 이를 기반으로 기준이 되는 손실률보다 측정된 값이 높으면 재전송 지속성을 증가시키고 낮으면 줄이는 방식을 제안하였다. [3,5]은 Markov 모델을 통해 ARQ와 TCP간의 동작을 분석을 통해 ARQ의 재전송 지속성 값을 결정하였으며, [4]의 경우 UMTS환경에서 RLC와 TCP 간의 interaction을 통한 방법과 이를 통해 RLC의 parameter 설정에 대한 가이드를 제시하였다. 하지만 이러한 기존의 연구¹⁻⁵⁾들 경우 ARQ가 무선 링크에서 발생하는 패킷의 error(loss)에 의한 패킷 손실률을 계산하고 이를 기반으로 패킷 손실률이 증가하면 재전송 지속성을 증가시키고 이에 반하면 반대로 재전송 지속성을 감소시키는 동작을 통해 전체 전송의 성능을 향상시킬 수 있다는 것을 보였다. 하지만 대부분의 연구들의 취하는 방식들은 재전송의 지속성을 관리하는데 있어서 패킷 손실을 위주로 동작을 수행한다. 이러한 방식은 패킷 손실률을 줄이고 상위 layer에 무선 링크에서의 패킷 손실을 보완함으로써 성능 이득을 얻을 수는 있지만 반대로 패킷 손실을 복구하는데 발생하는 delay로 인해 발생하는 문제들은 해결하지 못한다. 만약 패킷 손실률의 증가로 인해 과도한 재전송이 무선 링크에서 발생할 경우 복구로 인한 delay로 인해 상위 layer에 문제를 발생시킬 수가 있다. 이러한 문제 중 하나로는 과도한 재전송으로 인한 TCP 단에서의 스푸리어스 타임아웃과 같은 문제

를 발생 시킬 수 있다⁷⁾. 이후 단순히 패킷 손실률을 넘어 링크의 delay 또한 고려하여 재전송 지속성을 고려해야 한다는 연구들^{7,8)}이 진행되어 왔다.

그럼에도 불구하고 이러한 기존의 연구들은 하나의 무선 링크를 가정한 환경에서의 문제 해결 방법을 제안해 왔다. 하나의 무선링크만을 가진 환경의 경우 이러한 방법을 통해서 문제를 해결할 수 있지만 전체 경로에서 하나의 무선 구간이 아닌 다중의 무선 구간이 될 경우에는 상황이 달라진다. 다중의 무선 구간의 경우 여러 무선 구간들이 전송 경로에 모두 영향을 미치게 되고 결국 하나의 전송 경로에서 발생하는 패킷 손실률과 전송 지연의 문제가 더 가속화 시킬 수가 있다. 따라서 이러한 다중 무선 환경에서 기존의 패킷 손실률에 따라 재전송 지속성을 조절하게 되면 더 큰 delay를 발생하게 되어 기존의 문제를 더 악화시킬 수가 있다. 또한 delay를 고려한 지속성 방법을 사용할 지라도 전체 무선 링크 성능을 고려한 동작이 아닌 자신의 무선링크만을 이용해서 사용하는 것은 전체 무선 링크 delay 성능을 향상 보장하지 못한다. 따라서 다중 무선 링크 환경에서 전체 성능을 향상시키기 위해서는 ARQ가 전체 무선 링크에 대한 성능을 인지하고 있어야 하고 이에 따라 적합한 재전송 지속성 동작을 수행해야 한다. 이러한 동작을 수행하기 위해서는 ARQ간의 협력적인 정보 교환의 방법이 필요하고 이에 따른 재전송 지속성 관리 방법이 필요하다.

현재 무선 멀티홉 환경에서 제안된 ARQ에 관련된 연구들은 다음과 같다. 무선 멀티홉 환경에서 ARQ protocol의 성능에 대한 분석적인 연구^{10,11)}와 무선 멀티홉 환경에서 손실 복구 방식에 관한 연구⁹⁾가 있다. 분석적 연구는 멀티홉 환경에서 ARQ 프로토콜 분석을 통한 성능 모델링을 제안하였고 손실 복구 방식에 관한 연구는 재전송 지속성 관리 방안이 아닌 다중의 무선 구간 중에서 프레임의 손실이 일어난 무선 구간을 찾고 그에 대한 복구 방법을 제안을 통해 성능을 향상 시키는 연구들이 제안되어 왔다⁹⁾. 이러한 방식들은 무선 구간의 상태에 따라 전체 성능을 만족시키는 것이 아닌 무선 멀티 홉 환경에서의 성능 모델링이나 손실 복구 성능을 향상에 초점을 둔다는 것에 본 논문과 차이가 있다.

III. 무선 멀티 홉 환경에서의 재전송 지속성 관리 기법

제안된 방법 주된 목적은 각각의 ARQ가 전체 무선 링크의 요구조건을 만족하면서 최대한의 성능을

제공하기 위해 자신의 재전송 지속성을 자신의 무선 링크 상태와 다른 무선 링크 상태에 따라서 조절하는 데 있다. 제안된 방법은 크게 2가지 동작으로 구성되어 있다. 첫 번째는 ARQ들이 자신의 무선 링크 상태를 측정하고 이 정보를 다른 ARQ들과 교환함으로써 자신의 정보뿐만 아니라 다른 링크들의 상태를 인지하는 동작이다. 이후 수집된 정보를 가지고 요구되는 요구조건을 만족시키면서 전체 성능을 향상시키기 위해 재전송 지속성 관리 기법을 수행하게 된다.

3.1 무선 병목 링크와 전체 무선 링크 성능을 인 지하기 위한 ARQ간의 정보 교환 방법

본 절에서 제안하려는 방법의 목적은 무선 링크 중 에서 요구조건을 만족시키기 위한 또는 전체 성능을 향상에 가장 효율적인 링크를 찾아내기 위한 것이다. 일반적으로 전체 경로의 throughput의 경우 병목 링크의 성능에 의해 결정되어지고 또한 queuing delay와 같이 전송 지연으로 인해 발생하는 추가적인 delay는 병목 링크의 delay에서 가장 많이 발생되어 진다. 따라서 전체 성능을 향상시키기 위해서는 병목 링크부터 성능을 향상 시키는 것이 적합한 방법이다. 하지만 다중 무선 링크의 상황의 경우, 각각의 무선 링크의 상황에 따라 병목 링크가 바뀔 수 있다. 따라서 무선 링크의 상황에 따라 병목 링크를 찾아내는 방법이 필요하다.

무선 병목 링크를 찾기 위한 전체적인 동작은 그림 1과 같다. 각각의 ARQ들은 자신의 데이터 프레임 안에 자신의 링크 정보를 다음 ARQ에게 전달하게 되고 이후 다음 절의 재전송 지속성 관리 기법을 통해서 피드백 정보를 다시 정보를 준 ARQ에게 전송함으로써 서로간의 무선 링크 정보를 교환하게 된다. 자신이 전달하는 링크 정보는 그림 2와 같다.

구체적인 정보 교환 방법은 다음과 같다. 먼저 각각의 ARQ들은 자신의 링크 delay(Di)와 ARQ 재전송 지속성(RP)의 값을 측정한다. 자신의 링크 delay(Di)

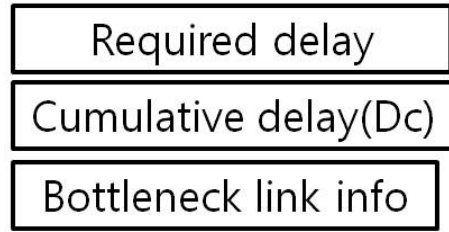


그림 2. 교환하는 링크 정보
Fig 2. exchange link information between ARQ

는 링크의 queuing delay만을 고려한다. 왜냐하면 propagation delay는 queuing delay에 비해 대부분 무시할만하고 또한 실제로 링크 delay 중에서 queuing delay 만이 조절이 가능한 링크의 delay이기 때문이다. 이후 ARQ들은 자신이 가지고 있는 무선 링크 정보를 다음 무선 링크에 있는 ARQ에게 전달한다. 전달되는 정보는 그림 2와 같다. 무선 링크에 요구 되는 delay(Dr)는 전체 경로 delay가 요구 delay를 만족하게 하기 위해 사용되어 지고 중복된 링크 delay(Dc)는 전체 무선 링크의 delay 성능을 예측하기 위해 사용된다. 이후 자신이 알고 있는 무선 병목 링크와 전송률 병목 링크를 선택하고 이에 대한 정보를 전달한다. 이를 선택하는 방법은 뒤에 다시 설명한다. 전송을 시작하는 ARQ의 경우 무선 병목 링크와 전송률 병목 링크 모두 자신의 링크로 설정한다. 전달되는 링크 정보들은 지연 병목 링크 넘버(Nd) 지연 병목 링크의 RP(RPd), 지연 병목 링크의 Di(Did), 전송률 병목 링크 넘버(Nt), 전송률 병목 링크의 RP(RPt) 그리고 전송률 병목링크의 Di(Dit)로 구성되어 있다. 이러한 정보들은 ARQ의 data frame를 전송할 때마다 함께 포함하여 전송되어진다.

링크 정보가 다음 무선 링크에 있는 ARQ에게 전달이 되었으면 다음 ARQ는 제일 먼저 받은 정보를 가지고 중복된 링크 delay(Dc) 를 계산하게 된다. 중복된 링크 delay는 식1과 같이 자신의 링크 delay(Di)

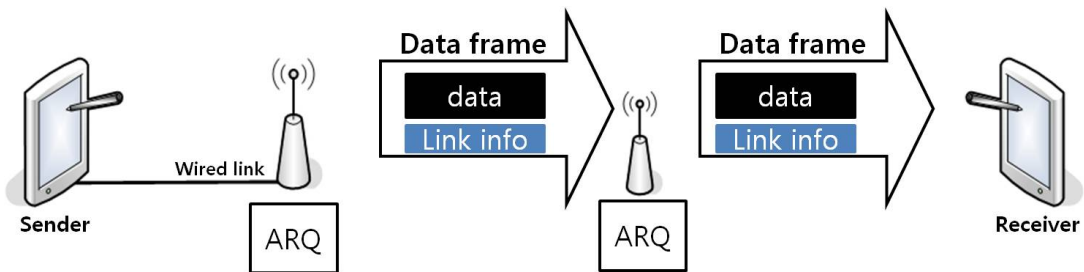


그림 1. 제안된 ARQ간의 정보 교환 방법.
Fig. 1. proposed intercommunication method.

와 받은 링크 정보에 있는 중복된 링크 delay의 값 (Drc)을 더한 값으로 계산한다.

$$Dc = Drc + Di \quad (1)$$

이후 받은 정보를 가지고 전체 무선링크에서의 병목 무선링크를 결정하게 된다. 무선 병목 링크는 지연 병목 링크와 전송률 병목 링크, 2가지로 구성 되어있다. 병목 링크는 링크 delay와 링크 재전송 지속성 2가지를 통해 결정 되어진다. 이 중에 링크 재전송 지속성을 먼저 고려하여 병목 링크를 정하는데 그 이유는 결국 링크 재전송 지속성의 요소가 링크의 성능을 변경 할 수 있기 때문이다. 따라서 링크의 정보 중에서 가장 높은 지속성을 가지고 있는 링크를 지연 병목 링크로, 반대로 가장 적은 지속성을 가지고 있는 링크는 전송률 병목 링크로 결정한다. 만약 지속성이 같은

경우 지연 병목 링크는 링크 delay의 값이 가장 큰 링크를 전송률 병목 링크의 경우 가장 작은 링크를 결정하게 된다. 무선 병목 링크를 다 결정하고 난 이후, ARQ는 계산한 중복된 링크와 결정한 무선 병목 링크 정보들을 데이터 frame에 실어 다시 다음 ARQ로 전송하게 된다. 정보교환 방법에 대한 전체적인 수도코드는 그림 3에서 확인 할 수 있다.

3.2 무선 멀티 홉 환경에서의 재전송 지속성 관리 기법

협력적 재전송 지속성 관리 기법의 주된 목적은 전체 무선 구간에서 요구되는 delay(Dr)를 만족시키면서 최대한의 전송률(throughput) 성능을 가지게 하는 것이다. 이를 수행하기 위해서 제안된 방법은 요구되는 delay를 넘어서지 않으면서 재전송 지속성을 최대화하기 위해 동작한다. 이를 만족시키기 위해 모든 ARQ들은 요구되는 delay를 충분히 만족할 수 있는 경우 재전송 지속성을 증가시키고 요구되는 delay를 만족시키지 못하는 경우 재전송 지속성을 줄이게 된다.

먼저 전송 구간의 요구되는 delay를 만족시키기 위해서, 제안된 방법은 ARQ에서 Dc가 요구되는 delay를 초과하는 경우 만에 동작이 수행된다. ARQ는 위의 절에서 설명한 협력적 방법을 통해 Dc를 계산하고 이 값이 Dr를 넘게 되면 delay control ACK를 만들어서 전송하게 된다. delay control ACK는 자신의 링크에서 결정한 지연 병목 링크의 정보를 ARQ의 ACK에 실어서 뒤로 전달하게 된다. 만약 자신이 지연 병목 링크라면 지연 병목 링크의 정보를 전달하지 않고 바로 자신의 링크의 재전송 지속성을 줄이게 되고 변화된 정보를 업데이트 하게 된다. 이와 관련된 동작의 flowchart는 그림 4에서 볼 수 있다. delay control ACK는 전체 무선 구간의 delay 성능이 Dr를 만족시키지 못한다는 의미와 delay control ACK가 나타내는 링크가 지연 병목링크로써 무선 성능을 향상 시켜야 한다는 의미를 가지고 있다. delay control ACK에서는 지연 병목 링크를 나타내기 위해 무선 링크 번호,RP,Di 의 값을 가지게 된다. 이렇게 전송된 delay control ACK를 뒤의 ARQ가 받게 된 경우 delay control ACK가 나타내는 링크가 자신의 링크인지 아닌지 판단하게 되고 자신의 링크가 아닌 경우 다시 자신의 뒤의 ARQ에게 delay control ACK를 전송하게 된다. 이 동작은 delay control ACK가 지연 병목 링크까지 도착할 때 까지 반복 된다. 이후 delay control ACK가 지연 병목 링크에 도착하게 되면 ARQ는 자신의 RP 값과 control delay ACK의 RP의 값을 비교

```

When data arrives in ARQ.
    update required delay(Dr), cumulative delay(Dc),
    link delay(Di) and retransmission persistence(RP)
    if there is no previous link information then
        Dc=Di
    end if
    if ARQ has other link information then
        Dc= Di+Drc // update Dc
        if RP > RPd then
            RPd=RP
            Nd = current link
        else if ( RP == RPd ) then
            if ( Di > Did ) then
                RPd=RP
                Nd = current link;
            end if
        end if
        if RP < RPt then
            RPt=RP
            Nt = current link
        else if ( RP == RPt ) then
            if ( Di < Dit ) then
                RPt=RP
                Nt = current link
            end if
        end if
    else
        Nd=Nt=current link
        RPd=RPt=RP
        Did=Dit=Di
    end if
    transmit the data with link info.
    
```

그림 3. 제안된 ARQ간의 정보 교환 방법의 수도코드
 Fig. 3. pseudocode of proposed intercommunication between ARQ

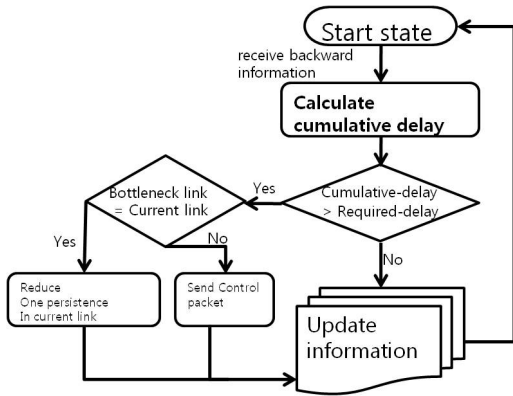
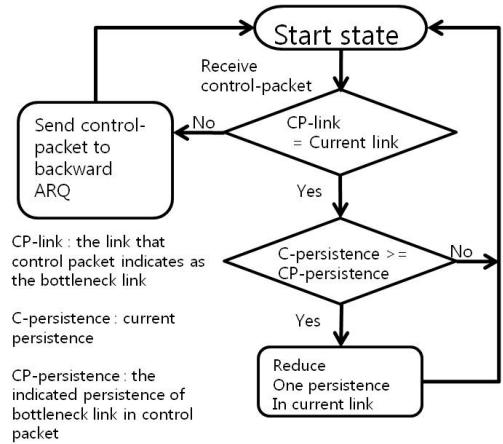


그림 4. 자신이 지연 병목 링크일 경우의 제안된 방법의 동작 순서도
 Fig. 4. flowchart of proposed scheme when proposed scheme is operated when received backward link information.

하게 되고 자신의 RP값이 delay control ACK의 RP의 값이 크거나 같은 경우에만 자신의 링크의 재전송 지속성을 1씩 줄이게 된다. 만약 delay control ACK의 RP가 자신의 값보다 작은 경우에는 이 delay control ACK를 무시하게 된다. 이렇게 무시하는 이유는 현재 상황이 아닌 이전의 상황의 값으로 인해 재전송의 변화의 반복을 최소화하기 위함이다. 이에 대한 동작 순서도는 그림 5에서 확인 할 수 있다. 위와 같은 동작으로 인해, 결국 지연 병목 구간의 링크부터 재전송 지속성을 감소시켜 가면서 전체 무선 구간이 Dr를 만족시킬 때 까지 반복해서 동작하게 된다.

위와는 반대로 제 Dr를 만족시키면서 전체 구간의 전송률을 최대화하기 위해 제안된 방법은 다음과 같이 동작한다. 먼저 전송률을 증가시키기 위한 delay 문턱을 결정한다. 이후 전체 구간의 delay가 delay 문턱을 넘지 않는 경우 전송률을 높이기 위해 충분한 delay 성능을 가진다고 판단하여, 재전송 지속성을 올리게 된다. 먼저 위의 delay control ACK와는 반대로 링크 정보를 받은 후 Dc를 측정하게 되고 이 값이 delay 문턱 보다 작은 경우, throughput control ACK를 뒤의 링크로 보내게 된다. throughput control ACK도 delay control ACK와 같이 무선 링크 번호, RP, Di의 정보를 가지게 된다. 이 ACK에는 전송률 병목 링크에 대한 정보를 기입하게 되고, 만약 자신이 전송률 병목 링크가 된다면 이 ACK를 보내지 않고 자신의 재전송 지속성을 1 증가시키게 된다. 이후 전송된 throughput control ACK를 받은 경우, 자신이 전송률 병목 링크가 아니면 다시 자신의 뒤의 링크에게 전송하게 되고 이 동작은 이 ACK가 전송률 병목 링크에



CP-link : the link that control packet indicates as the bottleneck link
 C-persistence : current persistence
 CP-persistence : the indicated persistence of bottleneck link in control packet

그림 5. delay control ACK를 받았을 때 제안된 방법의 동작 순서도
 Fig. 5. flowchart of proposed scheme when delay control ACK is arrived.

도착할 때까지 반복 된다. 이후 전송률 병목 링크가 이 ACK를 받게 되면 자신의 링크의 RP와 ACK의 RP를 비교하게 되고 자신의 RP가 ACK의 RP보다 작거나 같게 되면 자신의 재전송 지속성을 1씩 증가시키게 된다. 위와 같은 동작을 수행함으로써 제안된 방법은 요구되는 Dr를 만족시키면서 재전송 지속성을 증가하게 함으로 최대한의 전송률의 성능을 보장하게 한다.

IV. 시뮬레이션 및 성능 평가

4.1 시뮬레이션 환경

본 장에서는 제안된 방법의 성능을 평가하기 위한 시뮬레이션 환경을 설명한다. OPNET 14.5 버전을 이용하여 시뮬레이션 환경을 구축하였고 시뮬레이션의 환경 구조는 그림 6과 같다. 먼저 서버와 Base Station(BS) 사이의 구간은 유선구간으로 설정을 하고 BS부터 클라이언트 부분까지는 무선구간으로 설정되어 있다. 무선 구간은 BS 와 relay 사이의 무선 구간과 relay와 클라이언트 사이의 무선 구간으로 구성되어 있다. 각각의 무선 구간에는 ARQ가 사용되었다. 이러한 환경에서 제안된 방법은 무선 구간에 있는 각각의 ARQ에서 동작을 하게 하였다. 성능의 비교를 위해 기존의 ARQ의 경우 재전송의 지속성을 고정된 값(2,5)을 가지고 동작을 수행하였다. 어플리케이션으로 CBR traffic을 사용하였고 어플리케이션 packet size는 1024byte이다. transport layer는 TCP를 사용하였고 SACK option이 사용되어졌다. link layer에서

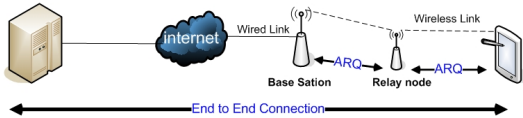


그림 6. 시뮬레이션 환경
Fig. 6. simulation environment

는 앞서 언급한대로 ARQ가 사용되었으며, 사용된 ARQ는 가장 기본이 되는 simple HARQ를 사용하였다. 무선 구간에서 요구되는 delay는 100ms부터 200ms 까지 변화하면서 실험을 진행하였으며, delay 문턱의 경우 요구되는 delay의 반의 값을 사용하였다. 전체 무선 구간에서의 프레임 손실률(FER)을 일정하게 유지하면서 각각의 무선링크의 프레임 손실률이 변경되게 함으로써, 프레임 손실률에 따라 병목 링크가 바뀔 수 있게 실험을 진행하였다. 성능 비교를 위해 제안된 방법은 최대 재전송 지속성을 5까지 최하로 2까지 내릴 수 있게 설정하였다. 나머지 시뮬레이션 파라미터 들은 표 1에 정리하였다.

표 1. 시뮬레이션 파라미터
Table 1. Simulation parameter

parameters	Value
application	CBR traffic, packet size = 1024byte
transport layer	TCP with SACK option
Link layer	simple HARQ, transmitter queue size = 200 frame,
required delay	100ms, 200ms
wired link	100Mbps, 1 ms
wireless link	2Mbps, 10ms
frame error rate(FER)	Total end-to-end FER = 0.2 each FER in wireless link is variable
retransmission persistence in original ARQ	2 , 5

4.2 시뮬레이션 결과 및 분석

그림 7은 요구되는 delay 성능을 0.2s로 가정하였을 때의 전체 무선 링크의 평균 delay 성능을 나타낸 것이다. 그림에서 persistence 2로 나타낸 결과는 재전송 지속성을 2로 고정한 기존의 ARQ 결과이고 persistence 5로 나타낸 결과를 재전송 지속성을 5로 고정한 ARQ의 결과이다. 결과를 확인해보면 persistence 2의 결과가 delay의 성능이 가장 좋은 것을 알 수 있다. 이러한 이유는 가장 적은 재전송 지속

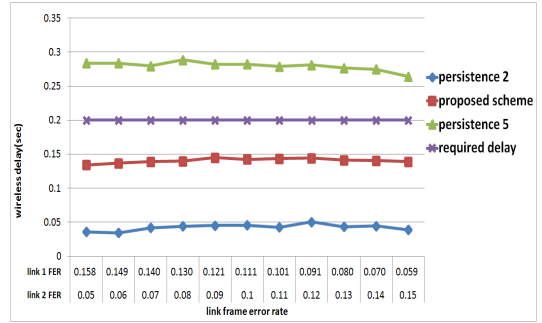


그림 7. 요구되는 delay가 0.2s 일 때 평균 wireless link delay 성능
Fig. 7. average wireless delay when required delay = 0.2s

성을 가지고 있기 때문에 FER로 인해 발생하는 delay가 가장 적기 때문이다. 이 경우 요구되는 delay의 성능을 만족시킬 수 있지만 전송률의 성능이 저하되게 된다. 반대로 persistence 5 경우 가장 delay 성능이 안 좋은 것을 확인 할 수 있다. 왜냐하면 가장 높은 재전송 지속성을 가지고 있기 때문에 중복된 프레임 손실 복구로 인해 많은 복구 시간을 소요하기 때문이다. 비록 더 신뢰성 있는 복구로 인해 전송률의 성능은 향상시킬 수 있지만 제안된 요구조건을 만족시키지는 못한다. 결과에서 보는 바와 같이, 제안된 방법은 요구되는 delay와 persistence 2 사이의 결과를 가진다. 제안된 방법의 경우 전체 무선 경로가 요구되는 delay를 넘는 경우 무선 경로 delay가 요구되는 delay보다 낮아질 때 까지 재전송 지속성을 낮추기 때문에 평균 delay는 요구되는 delay를 넘지 않게 된다.

그림 8은 요구되는 delay 성능을 0.2s로 가정하였을 때 전체 무선 링크의 평균 전송률 성능을 나타낸 것이다. 평균 전송률의 결과는 delay 결과와는 반대의 결과를 얻게 된다. 먼저 persistence 2의 경우 가장 낮은 평균 전송률을 가지는 것을 확인 할 수 있다. 이러한 이유는 앞서서 얘기한 것처럼 충분히 프레임 손실

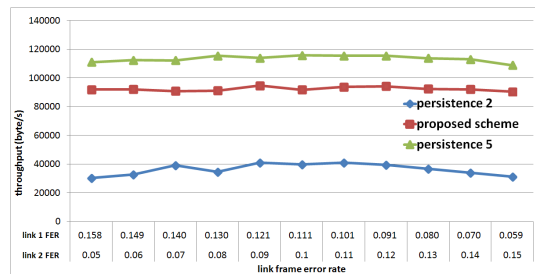


그림 8. 요구되는 delay가 0.2s 일 때 평균 전송률 성능
Fig. 8. average throughput when required delay = 0.2s

을 복구하지 못하기 때문이다. 반대로 persistence 5의 경우에는 많은 프레임의 손실을 복구 할 수 있기 때문에 링크의 프레임 손실을 줄여줌으로써 높은 전송률을 가지는 것을 확인 할 수 있다. 제안된 방법의 경우 persistence 5의 경우보다 적은 전송률을 가지지만 persistence 2보다 더 높은 전송률 성능을 가지는 것을 확인 할 수 있다.

그림 7, 8의 결과를 종합해 보면 고정된 재전송 지속성 방법을 사용할 경우 전송률의 성능 이득 또는 전송 delay의 이득에 따른 trade off 관계가 성립하게 된다. 제안된 방법의 경우 요구되는 delay를 만족시키면서, 즉 delay 성능을 향상시키지만 그에 따른 전송률 성능의 저하를 최대한 방지하는 것을 확인 할 수 있다.

그림 9, 10는 위의 상황에서 요구되는 delay를 변경했을 때의 결과이다. 그림 9은 요구되는 delay가 0.1 일 때 평균 wireless link delay를 나타내고 그림 10는 그에 따른 전송률 그래프를 나타낸다. 그림 9의 delay 성능을 보면 기존의 ARQ의 방법 달리 제안되는 방법은 요구되는 delay가 변하게 되면 그에 따라 전체 무선 링크의 delay 성능이 변하게 된다. 요구되는 delay가 내려감에 따라 제안된 방법 또한 delay 성능을 향

상시킴으로 요구되는 delay를 만족시키는 것을 확인할 수 있다. 기존의 ARQ 방법은 이전과 똑같은 delay 성능을 가지게 된다. 이에 따른 평균 전송률은 그림 10에서 보는바와 같다. 제안된 방법은 이전의 상황보다 전송률의 성능이 감소하였다. 이렇게 성능이 감소한 이유는 딜레이의 성능의 향상을 위해 재전송 지속성을 감소시켰기 때문이다. 하지만 성능이 감소하였다고 하더라도 delay 성능향상의 비율에 비해 전송률의 성능 감소를 최소화 하는 것을 확인할 수 있다. 이렇게 성능 감소를 최소화 할 수 있는 건 delay 성능 향상 안에서 최대한의 재전송 지속성을 유지하기 때문이다.

그림 11, 12은 반대로 요구되는 delay를 높였을 경우의 결과이다. 그림 12에서 보는 바와 같이 제안된 방법은 persistence가 5인 경우와 같은 성능을 보이는 것을 확인할 수 있다. 이렇게 같은 성능을 가지는 경우는 그림11에서와 같이 요구되는 delay의 조건이 충분해서 최대한 재전송 지속성을 올려 동작되기 때문이다. 또한 이 결과를 통해 제안된 방법 통해서도 기존의 방법과 거의 유사한 성능을 얻을 수 있기 때문에 제안된 방법으로 인한 오버헤드가 무시할만한 수준이

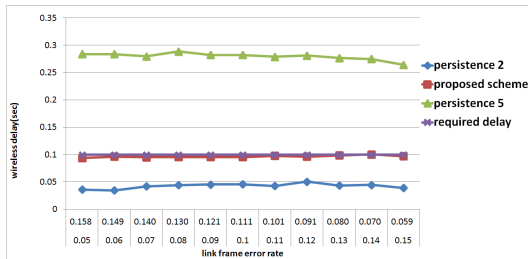


그림 9. 요구되는 delay가 0.1s 일 때 평균 wireless link delay 성능
Fig. 9. average wireless delay when required delay = 0.1s

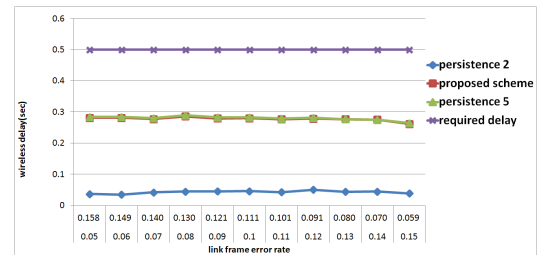


그림 11. 요구되는 delay가 0.5s 일 때 평균 wireless link delay 성능
Fig. 11. average wireless delay when required delay = 0.5s

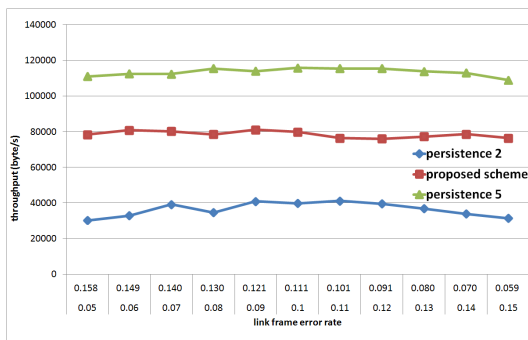


그림 10. 요구되는 delay가 0.1s 일 때 평균 전송률 성능
Fig. 10. average throughput when required delay = 0.1s

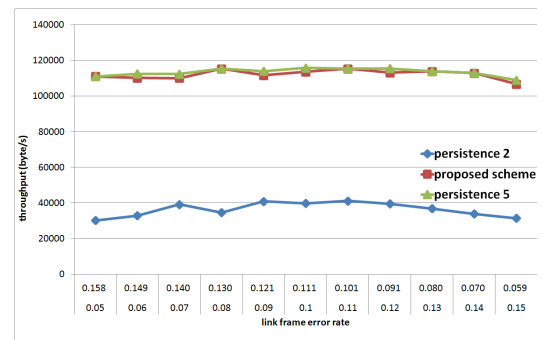


그림 12. 요구되는 delay가 0.5s 일 때 평균 전송률 성능
Fig. 12. average throughput when required delay = 0.5s

라는 것을 확인 할 수 있다.

위의 결과들은 종합해보면, 기존의 고정된 방법과는 달리 제안된 방법은 요구되는 delay 조건을 만족시키면서 최대한의 전송률 성능을 향상시키는 것을 확인 할 수 있다. 그리고 요구되는 delay 조건이 달라짐에 따라 전송률의 성능의 감소는 존재하지만 전체 무선경로의 delay 성능의 향상으로 인해 요구되는 delay 조건을 만족시키는 것을 확인 할 수 있었다. 따라서 제안된 방법은 링크의 프레임 손실율의 변화에 따라 그리고 요구되는 delay 조건을 만족시키기 위해 링크의 재전송 지속성을 변화시켜가면서 전체 무선 경로의 성능을 조절하는 것을 확인 할 수 있다. 또한 기존의 방법과 동일한 상황에서의 동작을 통해 제안된 방법의 오버헤드가 그리 크지 않은 것을 확인 할 수 있다. 또한 실제로 무선 구간이 늘어나는 경우 무선구간의 증가로 인해 발생하는 딜레이의 증가하게 될 것이고 이에 따라 전체 요구조건을 만족시키기 위해 각각의 ARQ의 재전송 지속성이 줄어들 수 있다. 또한 어플리케이션의 트래픽이 변화할 경우 다음과 같은 영향이 발생할 수 있다. 트래픽이 적어지면 그만큼 전송되는 프레임의 주기가 길어짐으로 그에 따른 링크 정보 교환의 피드백 시간이 길어지게 되고 반대로 트래픽의 증가로 인해 전송 주기가 짧아진다면 그에 따라 정보 교환 피드백 시간이 줄어들 수 있다.

V. 결 론

본 논문은 다중 무선 홉 환경에서 전체 무선 경로의 성능을 보장하기 위해서 ARQ 간의 재전송 지속성 관리 기법에 대해 제안하였다. 제안된 방법은 ARQ간의 링크 정보 교환 방법을 통해서 무선 병목 링크를 인지하게 되고 이를 통해 요구되는 전체 무선 링크 delay 요구조건을 만족시키면서 전송률 성능 향상을 위한 재전송 지속성을 변화하였다. 시뮬레이션 결과를 통해 제안된 방법은 제안된 요구 delay 조건에 따라 재전송 지속성을 변화시켜 가면서 전체 무선 구간에서 요구되는 delay 성능을 만족시키면서 delay 성능의 향상으로 인한 전송률 성능 저하를 최소화 하였다. 또한 요구되는 delay 조건이 변화됨에 따라 제안된 방법은 그에 적응하는 것을 확인 할 수 있었다. 제안된 방법을 통해 다중 무선 구간에서도 요구되는 delay을 보장이 가능하게 될 것이다. 추후에는 좀 더 다양한 실험 환경에 따른 성능 평가와 추가적인 알고리즘 분석을 통해 성능을 더 향상시키기 위한 연구를 추가적으로 수행할 예정이다.

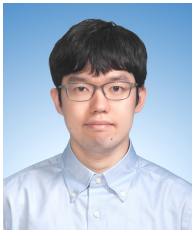
References

- [1] C. F. Chiasserini and M. Meo, "Modeling interactions between link layer and transport layer in wireless networks," *The 12th IEEE Int. Symp. PIMRC 2001*, San Diego, USA, Sep. 2001.
- [2] F. Vaciraca, A. D. Vendictis, and A. Baiocchi, "Investigating interactions between ARQ mechanisms and TCP over wireless links," in *Proc. of European Wirel.*, Barcellona, Spain, Feb. 2004.
- [3] A. Mehta, D. Kagaris, and R. Viswanathan, "Throughput performance of an adaptive ARQ scheme in rayleigh fading channels," *IEEE Trans. Wirel. Commun.*, vol. 5, no. 1, Jan. 2006.
- [4] C.F. Chiasserini and M. Meo, "A recon-gurable protocol setting to improve TCP over wireless," *IEEE Trans. Veh. Technol.*, vol. 51, no. 6, Nov. 2002.
- [5] J. J. Alcaraz, F. Cerdan, and J. Garcia-Haro, "Optimizing TCP and RLC interaction in the UMTS radio access network," *IEEE Network*, vol. 20, no. 2, Mar. 2006.
- [6] J. G. Kim and M. M. Krunz, "Delay analysis of selective repeat ARQ for a markovian source over a wireless channel," *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, Sept. 2000.
- [7] V. Subramanian, K. K. Ramakrishnan and S. Kalyanaraman, "Experimental study of link and transport protocols in interference-prone wireless LAN environments," *IEEE COMSNETS*, 2009.
- [8] J. Han, B. Kim, and J. Lee. "TCP-friendly retransmission persistence management for SR-ARQ protocols," *IEICE Trans. Commun.*, vol. E92-B, no. 10, pp. 3243-3246, Oct. 2009.
- [9] R. R. C. Bikram, N. Charbonneau, and V. M. Vokkarane, "Multi-layer loss recovery in TCP over optical burst-switched networks," *Photonic Network Commun.*, vol. 21, no. 2, pp. 158-169, Apr. 2011.
- [10] T. Issariyakul and E. Hossain, "Analysis of

end-to-end performance in a multi-hop wireless network for different hop-level ARQ policies," *IEEE GLOBECOM*, vol. 5, pp. 3022-3026, Nov.-Dec. 2004.

- [11] L. Long, E. Hossain, "An analytical model for ARQ cooperative diversity in multi-hop wireless networks," *IEEE Trans. Wirel. Commun.*, vol. 7, no. 5, pp. 1786-1791, May 2008.

오 봉 환 (Bong-Hwan Oh)



2009년 2월 : 연세대학교 전기 전자공학부
 2011년 2월 : 연세대학교 전기 전자공학부 석사
 2011년 3월~현재 : 연세대학교 전기 전자공학부 박사과정
 <관심분야> Transport layer protocols, wireless multihop Networks

김 석 규 (Seog-gyu Kim)



1990년 2월 : 연세대학교 전자 공학과 졸업
 1992년 8월 : 연세대학교 전자 공학과 석사
 1997년 8월 : 연세대학교 전자 공학과 박사
 1997년 9월~2004년 3월 : SK 텔레콤 선임연구원

2004년 3월~2006년 3월 : 연세대학교 전기전자공학과 IT 연구단 연구 교수
 2006년 4월~현재 : 안동대학교 공과대학 정보통신공학과 부교수
 <관심분야> Wireless Sensor Network, 유,무선 통합망에서 QoS Architecture, TCP 성능 분석, 이동통신, IoT

이 재 용 (Jaiyong Lee)



1977년 2월 : 연세대학교 전자 공학과
 1984년 5월 : IOWA State University 공학석사
 1987년 5월 : IOWA State University 공학박사
 1987년 6월~1994년 8월 : 포항 공과대학 교수

1994년 5월~현재 : 연세대학교 전자공학과 교수
 <관심분야> Protocol Design for Wired/Wireless QoS Management, Ubiquitous Sensor Network, Wireless Multihop Network