
IPv6 헤더 압축에서의 에러 복구 방안

신병철* · 김민영* · 흥고르츨* · 이인성*

Error Recovery Schemes with IPv6 Header Compression

Brian B. Shinn* · Min-Yeong Kim* · Khongorzul D* · In-Sung Lee*

이 논문은 2004년도 한국과학재단 목적기초연구과제 (R01-2003-000-11620-0) 지원으로 수행되었습니다.

요 약

헤더 압축 기술은 기존의 일반적인 패킷 데이터의 비효율적인 오버 헤드를 줄이기 위한 방안으로 제안되었다. 특히 음성과 같은 실시간 미디어 스트림 경우에는 더욱 큰 오버헤드율을 보이게 되는데, 헤더 압축을 통해 보다 높은 대역폭 효율을 얻을 수 있다.

본 논문에서는 PHR (Periodic Header Refresh) 방법과 HR (Header Request) 방법의 두가지 에러 복구 기법에 관하여 각각 성능 분석을 실시하였다. 그 결과로 Overhead Rate, Bandwidth Gain, 그리고 Bandwidth Efficiency 등의 분석 결과가 제시되었다.

ABSTRACT

Header compression scheme is suggested as a solution to reduce the inefficient overhead of general packet stream data. Especially, it is shown that there are more overhead rate for real-time media stream links such as voice because of its short payload size, and it is possible to get higher bandwidth efficiency using the header compression scheme.

There are two kinds of error recovery in header compression such as Periodic Header Refresh (PHR) and Header Request (HR) schemes. In this paper, we analyze the performance of these two compression recovery schemes, and some results such as the overhead rate, bandwidth gain and bandwidth efficiency (BE) are presented.

키워드

Header compression, Error recovery, IPv6

I. 서 론

최근 멀티미디어 통신 서비스를 위한 많은 관심이 집중되고 있다. 특히 인터넷을 이용한 IP 기반의 멀티미디어 서비스가 그 영향력을 키워나가고 있다. 더구나 기존

회선교환 기반의 음성위주의 통신서비스는 점차 IP기반의 통신으로 변화하고 있는 추세이며, 기존 IPv4를 대체하는 IPv6[1]의 개발로 그 중요성은 더욱 확고해지고 있다.

그러나 실시간 데이터 패킷 스트림의 경우에, IP는 UDP/RTP와 결합하면 페이로드에 비해 큰 오버헤드를 가

지게 된다. 예를 들어 VoIP의 경우, 보통 data의 길이가 20 bytes 인데 비하면 40 바이트의 IPv4/UDP/RTP 헤더는 페이로드 사이즈에 비하여 오버헤드 크기가 2배 정도가 된다. 게다가 기본 헤더의 크기가 40바이트인 IPv6상에서 서비스가 된다면, 그 오버헤드는 더욱 커진다. 따라서 한 정적인 대역폭을 보다 효율적으로 제공하는 방안이 필요하게 되었다.

이를 해결하기 위해 VJHC [2], IPHC [3], ROHC [4], Compressed RTP [5] 등의 RFC 문서들이 제공되었으며, 이 문서들의 공통적인 특징은 기존 헤더를 압축하는 방안을 제안하고 있다. 헤더 압축 기법은 각 헤더 필드 중에서 공통된 값을 가지거나, 일정한 차이의 값을 가지는 필드들을 제거하여 헤더의 크기를 줄여서 보내는 기법이다. 이런 기법을 적용하기 위해 각 노드는 헤더를 압축하고 재조립하는 기능을 가져야 하며, 송신자와 수신자는 압축과 관련한 정보를 서로 공유하는 동기화가 이루어져야 한다. 만일 비트 에러나 패킷 손실이 발생하게 된다면, 압축 정보는 왜곡이 되어 적절한 재조립이 이루어지지 않게 되고 기존에 보내진 패킷들은 재조립 과정에서 모두 폐기된다. 이러한 에러로 인한 패킷 손실을 복구하기 위해 Periodic Header Refresh (PHR) 기법과 Header Request (HR)의 두가지 에러 복구 기법에 제시되었다.

PHR의 경우는 주기적으로 Full_Header를 보내어 패킷의 손실을 최대한 줄일 수 있다는 장점을 가지고 있으며, 패킷의 손실이 적어 보다 안정적인 서비스를 제공할 수 있다. 그러나 주기적으로 Full_Header를 보내는 관계로 HR에 비해서 효과적인 대역폭 절감을 기대할 수 없다는 점과, Full_Header를 보내는 주기 시점을 결정해야 하는 등의 문제점이 있다 [3].

HR 에러 복구 기법은 Sequence Number를 조사하여 이전에 받은 패킷의 번호와 비교하여 적합한 패킷인지 확인하여 만일 적합하지 않다면, 수신자는 송신자에게 "context_state" 라는 패킷을 보내고, 송신자는 "Full_Header" 를 보내어 송신자와 수신자 간의 압축 정보를 다시 갱신하게 된다. 이 경우는 만일 송신자와 수신자간의 피드백 시간이 긴 경우에 폐기되는 패킷이 많아지게 되어 패킷 손실율이 커지게 되고, 에러율이 높은 링크의 경우에는 패킷 손실이 많이 발생하여 송수신단간의 동기화 유지가 어렵게 된다 [6]. 이에 따라 본 논문에서는 PHR와 HR의 에러 복구 기법을 분석해 보고자 한다.

본 논문의 2장에서는 일반적인 헤더 압축 기법에 대해

알아본다. 3장에서는 제안된 헤더 압축 에러 복구 기법에 대해 살펴볼 것이며, 4장에서는 그에 따른 성능 분석이 어떻게 나왔는지 살펴본다. 마지막으로 5장에서 결론을 내고 논문을 마무리 하고자 한다.

II. 관련연구

현재 사용되고 있는 IP 프로토콜은 기본적으로 위의 상위 계층에 TCP (Transport Control Protocol)나 UDP (User Datagram Protocol), 그리고 VoIP와 같은 실시간 통신을 위한 RTP (Real Time Protocol)[7]을 사용하고 있다. 대부분의 실시간 서비스에서와 같이 헤더에 비해 데이터가 적은 서비스가 갖는 큰 오버헤드는 네트워크의 대역폭 효율을 저하시키는 원인이 된다. 이와 같이 데이터에 비하여 헤더가 큰 경우 헤더 압축 기술을 이용하면 많은 오버헤드를 줄일 수 있으며, 네트워크의 대역폭 효율을 높히는데 많은 도움이 될 수 있다. 이와 관련된 여러 헤더 압축 기법들이 제안되었으며, TCP 프로토콜의 헤더 압축 연구로는 VJHC [2], IPHC [3], ROHC [4] 등이 있다.

Degermark [3]는 연결이 처음 시작된 Slow-Start 시점에서 송수신단 사이에 full header packet 사이에 보낼 수 있는 압축헤더가 몇 개를 보낼 수 있는 기간이 2배씩 늘어난다는 사실과, 헤더 refresh 사이에 보낼 수 있는 압축된 헤더의 non-TCP(즉, UDP) 패킷의 개수 및 2개의 인접 full header 사이의 허용된 최대시간 등에 관하여 논의하고 있다.

VJHC 알고리즘을 개선하기 위하여 제안된 새로운 알고리즘을 살펴본다. Giovanardi [6] 등은 헤더 압축방법으로 2가지 방안을 제안하였다. 먼저 ABP-SCS(Adaptive Base Proposal with Static Compression Start) 방식에서는 헤더 베이스(header base)를 업데이트 한다는 송신단 메시지와 그것을 받았다는 수신단에서의 ACK를 교환하는 방식을 사용하면 양단 사이에 동기를 잃을 염려가 없다. 제안된 또 다른 한 가지인 ABP-DCS (Adaptive Base Proposal with Dynamic Compression Start)에서는 송신단이 비압축 패킷을 계속 보내다가 ACK 응답이 오면 비로소 압축 패킷을 보내는 방법이다. 이 2가지 방식이 기존의 VJHC [2] 이나 Degermark [20], Perkins [21], 또는 Calveras [22]가 제안한 방식보다 우수한 성능을 보여주고 있다.

III. 헤더 압축 에러 복구 기법

헤더 압축은 링크의 효율성을 높이기 위하여 여러 응용망에 적용될 수 있다. 높은 에러율과 지연율을 가지고 있는 위성 통신망이나 셀룰러 통신망, 그리고 저대역폭을 지원하는 PPP망이나 일반 전화망 등에도 적용될 수 있으며 비교적 대역폭이 높고 트래픽도 많은 WAN (Wide Area Network)의 경우에 적용될 수 있다. 이러한 여러 가지 응용망 중에서 최근 많이 사용되고 있는 802.11 무선랜 (WLAN)의 경우를 생각해 보고자 한다. WLAN에 있어서는 먼저 오버 헤드에 관련된 효율을 향상시켜야 할 필요가 있다. 아래의 그림 1은 802.11의 무선랜 환경에서의 에러 복구 개념을 보여주고 있다.

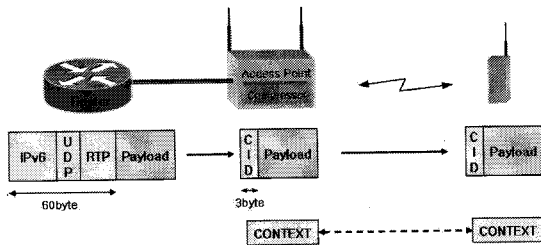


그림 1. 무선랜의 헤더 압축 환경

Fig. 1. Header Compression Environment of Wireless LAN

그림 1에서의 VoIP (Voice over IP)에서 사용되는 음성 패킷의 예를 들어보면, 여러 경로를 통해 수신단 쪽의 마지막 라우터를 통해 들어온 패킷은 IPv6/UDP/RTP의 경우에는 헤더가 60바이트로써 20바이트의 페이로드까지 합한 전체길이에서 약 75%의 오버헤드를 보이고 있다. 무선랜의 경우, 페이딩(Fading)이나 간섭 등으로 인한 높은 에러율과 손실율 등으로 헤더 압축 기법의 적용이 쉽지 않다. 또한 유선망에 비해 채널 상태의 변동 확률이 높아 송수신단 간의 동기화 유지가 쉽지 않다[11]. 따라서 802.11 무선랜 환경에서 컴프레서 (Compressor)와 디컴프레서 (Decompressor)는 패킷 에러나 손실에 대해서 강력한 동기화를 유지해야 하며 단일 Context를 유지하는 동기화가 어긋날 경우 빠른 에러 복구를 통해 폐기되는 패킷의 수를 최소화 시켜야 한다.

IV. 성능 분석

우선 패킷의 오버 헤드율과 그에 따르는 대역폭의 이득율을 비교하고자 한다. 먼저 VoIP 음성신호를 전송하기 위한 IPv6/UDP/RTP에서의 상황을 가정한다. 페이로드는 10에서 100바이트 사이로 한정하였으며 음성패킷의 샘플링 주파수는 8KHz로 하였다. IPv6/UDP/RTP는 옵션 헤더나 확장되는 헤더가 없이 기본적인 헤더만 사용된다고 가정한다. 따라서 이에 따르는 네트워크 계층까지의 헤더 크기는 60바이트라 가정할 수 있다. 또한 기본적인 헤더를 사용되기 때문에 최대 압축된 크기인 2바이트의 헤더 크기를 가진다고 가정할 수 있다. 먼저 아래에서 사용될 파라미터의 정의는 다음과 같다:

f_r : 소스 프레임 전송률

B_{WC} : 비압축 일반 패킷에서 요구되는 대역폭

B_C : 헤더 압축된 패킷에서 요구되는 대역폭

l_u : 일반 패킷의 헤더 길이

l_c : 헤더 압축된 패킷의 헤더 길이

B_{Gain} : 헤더 압축된 경우에 얻게 되는 대역폭 이득

$$B_{GainRate} = B_{Gain}/B_{WC} (=이득/비압축대역폭)$$

4.1. 페이로드에 따른 오버 헤드율 비교

기본적으로 오버헤드는 헤더의 크기와 헤더와 데이터를 합한 크기의 비율로 나타낼 수 있다. 일반적인 패킷에 대해 다음의 수식으로 오버 헤드율을 구할 수 있다.

Overhead

$$= \frac{Header\ Size}{Header\ Size + Payload\ Size} \times 100\% \quad (1)$$

기본적으로 압축되지 않았을 때의(noHC: no Header Compression) 패킷의 오버 헤드율과 HR 에러 복구 기법일 때의 오버 헤드율, 그리고 주기적으로 Full_Header를 보내는 PHR 에러 복구 기법에 관한 오버헤드율을 아래와 같이 파악하였다. 이 경우 링크의 에러는 고려되지 않았다.

OverheadRate (PHR)

$$= \frac{l_u + (N - 1)l_c}{l_u + (N - 1)l_c + N \cdot D} \quad (2 - 1)$$

$$OverheadRate (HR) = \frac{l_c}{l_u + D} \quad (2-2)$$

$$Overhead (noHC) = \frac{l_u}{l_u + D} \quad (2-3)$$

위의 식을 그래프로 그려보면 그림 2과 같다. 그림2에서 보는 바와 같이 압축하지 않은 경우 (noHC)보다 압축한 경우가 Overhead Rate가 낮은 것을 볼 수 있다. 또한 Refresh주기 N이 커질수록 (N=8에서 N=64로) PHR 경우의 오버 헤더율은 기존의 HR를 적용했을 경우와 비슷해지는 경향을 보이며, 페이로드의 크기가 커질수록 각각의 오버 헤더율은 크게 차이가 나지 않는 것을 볼 수 있다.

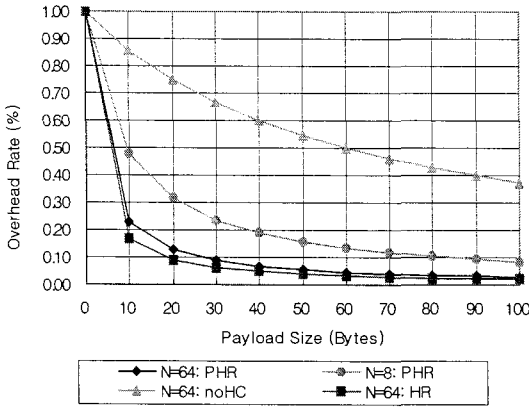


그림 2. 페이로드에 따른 오버 헤더율 비교
 ($P_b = 10^{-7}$, $P_u = 0.0000640$, $P_c = 0.0000176$)
 Fig. 2. Comparison of overhead rate as a function of payload
 ($P_b = 10^{-7}$, $P_u = 0.0000640$, $P_c = 0.0000176$)

4.2. 페이로드에 따른 대역폭 이득을 비교

다음에는 Bandwidth Gain 에 관하여 살펴보기로 한다. 먼저 압축되지 않은 패킷의 대역폭은

$$B_{WC} = (l_u + D) f_r$$

과 같이 되고 각 압축 방식에 의하여 얻을 수 있는 Bandwidth Gain은

$$B_{Gain}(HR) = (l_u - l_c) f_r$$

$$B_{Gain}(PHR) = (l_u - l_c) \frac{N-1}{N} f_r \quad (3)$$

와 같다. 최종적으로 Gain Rate는 $\frac{B_{Gain}}{B_{WC}}$ 으로 주어지므로

$$B_{GainRate}(PHR)(\%) = \frac{l_u - l_c}{l_u + D} \cdot \frac{N-1}{N} \quad (4)$$

$$B_{GainRate}(HR) = \frac{l_u - l_c}{l_u + D}$$

와 같이 유도된다. 그림 3에서는 위의 수식을 바탕으로 페이로드의 크기에 따라 대역폭 이득율을 비교하였다. 각 데이터를 보내기 위해 필요한 대역폭과 전체 헤더를 보내기 위한 대역폭의 비율로 파악될 수 있다. 그림 3에서 보는 바와 같이 페이로드가 커질수록 PHR, HR 경우 모두 대역폭 이득율은 감소한다. 또한 HR에서 제일 높은 이득율을 보이고 있다.

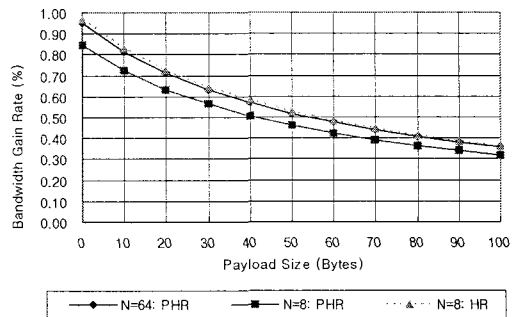


그림 3. 페이로드에 따른 대역폭 이득율 비교
 Fig. 3. Comparison of Bandwidth Gain Rate with payload

4.3. 비트 에러율에 따른 대역폭 효율 비교

대역폭 효율은 보내진 전체 패킷의 크기 중에서 실제 Decompressor가 받아 사용한 페이로드의 크기의 비율로 일반적으로 다음과 같이 나타낼 수 있다:

$$Bandwidth\ Efficiency\ (\%) = \frac{Received\ Payload\ Size}{Transmitted\ Packet\ Size} \times 100 \quad (5)$$

PHR에서는 먼저 프레임의 정의를 한다. 프레임의 맨 앞에서는 비압축 헤더가 한개 전송되고 그 뒤에 압축헤더가 (N-1)개 전송되어 전체적으로 N개의 패킷이 한 프레임 내에서 전송된다. 만약 (N-k-1)번째 패킷에서 전송에러가 발생하면 에러가 전파되어 그것을 포함한 그 이후에 있는 k개의 모든 패킷들이 압축해제가 되지 못하고 폐기되게 된다. 본 논문에서는 실제 상황에 맞는 모델링을 하기 위하여 다음과 같은 경우를 생각해 본다. 즉 헤더가 압축된 경우와 그렇지 않은 경우는 헤더의 길이에 있어서 매우 큰 차이가 나게 되고 따라서 헤더의 전송에러에도 큰 차이가 발생할 수 있다. 패킷 헤더의 길이를 고려하여 모델링해 보면, 확률 $P(N-k, k)$, 즉 N개의 패킷 중에서 (N-k)개는 수신단에서 올바르게 압축해제 되고 k개의 패킷 헤더가 전송오류로 인하여 압축해제가 되지 않을 확률은, 앞부분의 (N-k)개는 오류없이 전송되었으나, (N-k+1) 번째 헤더 전송에 에러가 발생한 경우로써 그 확률은 다음 식과 같이 3가지 경우로 주어진다.

$$P(N-k, k) = \begin{cases} (1-P_u)(1-P_c)^{N-1}, & k=0 \\ (1-P_u)(1-P_c)^{N-k-1}P_c, & k=1, 2, \dots, N-1 \\ P_u, & k=N \end{cases} \quad (6)$$

위의 식에서 사용된 파라미터의 정의와 구하는 방식은 다음과 같다.

P_u : probability of error for uncompressed packet

P_c : probability of error for compressed packet

$$P_u \cong P_b(l_u + D)$$

$$P_c \cong P_b(l_c + D)$$

위의 식(6)에서 첫째 항은 모두 성공적으로 전송되어 오류가 없는 경우이고, 중간 항은 맨 앞의 uncompressed 헤더와 2번째 패킷부터 (N-k)번째까지의 모두 (N-k-1)개의 compressed 패킷이 무사히 전송되고 있어서 (N-k+1)번째 압축패킷의 헤더에 에러가 발생하는 경우이다. 끝으로 셋째 항은 맨 먼저 전송되는 압축되지 않은 헤더에서 오류가 발생하여 하나의 패킷도 전송에 성공하지 못한 경우이다. 이 확률 식을 k에 관하여 0부터

N까지 합하면 1이 되는 것을 확인할 수 있다. 한 프레임 내에서 압축해제가 되지 않는 패킷의 평균 갯수 $N_e(PHR)$ 를 구하기 위해서는 식(6)으로부터

$$N_e(PHR) = \sum_{k=0}^N kP(N-k, k) \text{을 정리해보면,}$$

$$N_e(PHR) = \frac{P_c^N(1-P_u)}{1-P_c} + NP_u \quad (7)$$

과 같이 된다. 식(7)은 N개의 패킷을 전송하면서 발생한 것이기 때문에 이를 N으로 나누면 PEP (Packet Error Probability)가 다음과 같이 구해진다.

$$\begin{aligned} PEP(PHR) &= \frac{\text{전송에 성공한 패킷수}}{\text{전송된 전체 패킷수}} \\ &= \frac{N_e}{N} = \frac{(1-P_u)P_c^N}{N(1-P_c)} + P_u \end{aligned} \quad (8)$$

이어서 BE (Bandwidth Efficiency)를 검토하기 위하여 먼저 사용될 기호들을 다음과 같이 정의 한다:

l_s : Transmitted packet size including error packets

l_{rp} : Received payload size

위의 기호를 사용하면 BE는 다음과 같이 나타낼 수 있다:

$$BE(PHR) = \frac{l_{rp}}{l_s} \quad (9)$$

위식을 헤더 압축에 적용하려면 압축 시 줄어드는 헤더 길이와 전송 에러에 따른 재전송으로 인해 증가하는 전송 비트수 등을 고려하여야 한다. 한 프레임을 구성하는 N개의 패킷을 모두 보낼 때 그 전체길이는 payload 데이터의 길이 및 압축헤더와 비 압축 헤더의 길이를 모두 합하여야 한다. 이러한 기본 개념 하에서 PHR 경우에 있어서, Received payload size와 Transmitted packet size를 각각 l_{rp} 와 l_s 라고 하면,

$$\begin{aligned} l_{rp} &= \sum_{k=0}^N (N-k)D \cdot P(N-k, k) \\ &= D[N(1-P_u)(1-P_c)^{N-1} \\ &\quad + \frac{1-P_c}{1-2P_c} \left(\frac{P_c}{1-P_c} \right)^{N-1}] \end{aligned} \quad (10)$$

$$l_s = l_u + (N-1)l_c + ND$$

가 되고 BE(PHR)는

$$BE(PHR) = \frac{D[N(1-P_u)(1-P_c)^{N-1} + \frac{1-P_c}{1-2P_c}(\frac{P_c}{1-P_c})^{N-1}]}{l_u + (N-1)l_c + ND} \quad (11)$$

와 같이 된다.

다음은 HR의 경우를 아래와 같이 검토한다. 관찰 대상이 되는 임의의 Tagged Packet에 Error가 k 번 발생할 확률 $P_{hr}(k)$ 은

$$P_{hr}(k) = \begin{cases} (1-P_c) & , k=0 \\ P_c P_u^{k-1} (1-P_u) & , 1 \leq k \leq N \end{cases} \quad (12)$$

과 같이 된다. 여기서 N_r 은 허용된 최대 재전송 횟수를 나타낸다. 다음은 하나의 패킷을 전송하기 위하여 소요된 전체 비트수를 계산하여 보자. 실패한 횟수를 k 라고 할 때 전체 전송비트수는 예를 들면 다음과 같다:

$$\begin{aligned} k=0: & \quad l_c + D \\ k=1: & \quad (l_c + D) + \left\lceil \frac{a}{l_c + D} \right\rceil (l_c + D) + (l_u + D) \\ k=2: & \quad (l_c + D) + \left\lceil \frac{a}{l_c + D} \right\rceil (l_c + D) \\ & \quad + (l_u + D) + \left\lceil \frac{a}{l_u + D} \right\rceil (l_u + D) \\ & \quad + (l_u + D) \end{aligned} \quad (13)$$

위의 식을 $k=1$ 경우에 살펴보면, Tagged Packet의 압축 헤더를 포함한 데이터가 전송되고, 그 뒤에 Roundtrip Time 시간 동안에 전송되는 뒤에 오는 패킷 수 $\lceil a/(l_c + D) \rceil$ 를 계산하고(여기서 $\lceil x \rceil$ 는 x 보다 큰 최소 정수를 의미한다), 그 패킷 갯수에 압축 패킷 길이 $(l_c + D)$ 를 곱하여 더하고, 끝으로 에러로 인하여 재전송되는 패킷의 길이 $(l_u + D)$ 를 더하는 것으로 계산할 수 있다. 위의 식을 바탕으로 일반적으로 k 개의 연속된 에러가 발생시 전송되는 전체 비트수 $l_{hr}(k)$ 는 다음과 같이 주어진다:

$$l_{hr}(k) = \begin{cases} (l_c + D), & k=0 \\ (l_c + D) + \frac{|a|}{l_c + D} (l_c + D) \\ + (l_u + D) + \frac{|a|}{l_u + D} (l_u + D)(k-1) \\ + (l_u + D), & 1 \leq k \leq N_r \end{cases} \quad (14)$$

여기서 하나의 packet이 전송에 성공하기 위해서 소요되는 비트수 L 는 다음과 같다:

$$\begin{aligned} L &= \sum_{k=0}^{N_r} P_{hr}(k) l_{hr}(k) \\ &= (1-P_c)(l_c + D) \\ & \quad + \sum_{k=1}^{N_r} P_c P_u^{k-1} (1-P_u) l_{hr}(k) \end{aligned} \quad (15)$$

또한 하나의 패킷이 전송에 성공하기 위하여 발생하는 평균 전송오류 횟수 K 는 다음과 같이 정리된다.

$$\begin{aligned} K &= \sum_{k=0}^{N_r} k P_{hr}(k) \\ &= P_c \frac{1 - (1-N_r)P_u^{N_r} + N_r P_u^{N_r+1}}{1-P_u} \end{aligned} \quad (16)$$

지금까지의 풀이를 바탕으로 PEP와 BER를 구해보자. 전송에 성공하기 위해서는 처음에는 오류가 발생하더라도 마지막에는 전송이 성공하여야 하므로 전체 전송횟수는 K 보다 1만큼 커야 한다. 따라서 PEP와 BE는 다음과 같이 구할 수가 있다.

$$\begin{aligned} PEP(HR) &= \frac{\text{전송에 성공한 패킷수}}{\text{전송된 전체 패킷수}} = \frac{K}{K+1} \\ BE(HR) &= \frac{D}{L} \end{aligned} \quad (17)$$

위식을 근거로 관련 수치결과를 도출하여 보았다. 먼저, 그림4에서는 PEP 값을 Bit Error Rate (BER)의 함수로 그려 보았다. 그림4에서 보는 바와 같이 BER이 증가함에 따라 일반적으로 PEP도 증가함을 알 수 있다. PHR의 경우를 보면 HR 경우에 비하여 큰 값을 가진다. HR의 경우에는 $N_r=8$ 에서 $N_r=64$ 로 증가함에 따라 평균전송

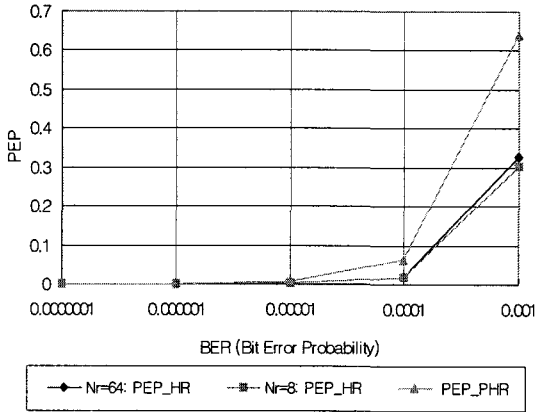


그림 4. BER의 함수로 표시된 PEP 값의 그래프.
Fig. 4. PEP values as a function of BER.

오류횟수 K 가 증가하고 그에 따라 PEP가 약간 낮아진다.

다음에는 그림5에 있는 BE를 살펴보기로 한다. HR의 경우가 PHR의 경우보다 큰 값을 보여주고 있다. PHR의 경우에는 BER이 낮으면 $N=64$ 의 경우가 $N=8$ 의 경우보다 더 높은 BE를 보여주고 있고, 반대로 BER이 높으면 $N=8$ 의 경우가 $N=64$ 의 경우보다 더 높은 BE를 보여주고 있다. 따라서 채널 오류율에 따라서 적합한 전송 오류 복구 방안을 취사선택할 필요가 있다고 판단된다.

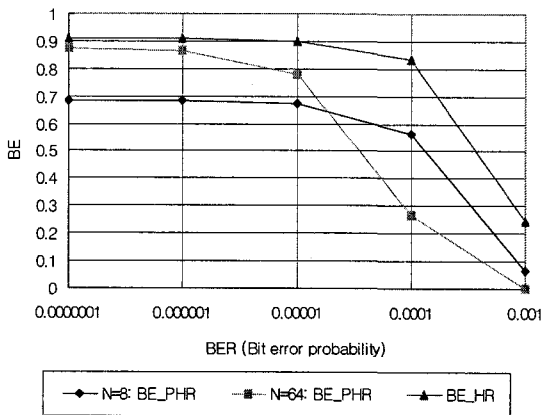


그림 5. BER의 함수로 표시된 Bandwidth Efficiency
Fig. 5. Bandwidth Efficiency value as a function of BER.

V. 결 론

본 논문에서는 기존 헤더 압축의 에러 복구 기법인 Periodic Header Refresh (PHR) 기법과 Header Request (HR) 기법의 특징을 파악하고, 이에 대한 성능을 헤더 길이를 고려하여 분석하여 보았다. 각 기법에 대한 오버헤드율, Bandwidth Gain Rate, PEP (Packet Error Probability) 등을 비교하였고, 그에 따르는 대역폭 효율 BE(Bandwidth Efficiency)을 구하였다.

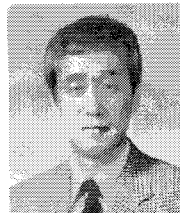
분석한 주요 결과로는, 전송 오류 복구 방안으로 PHR 기법을 채택하는 경우에 BE는 BER (Bit Error Probability)이 낮을 때에는 $N_r=64$ 가 $N_r=8$ 경우보다 더 높은 BE를 보여주고 있고, BER이 높으면 $N_r=8$ 의 경우가 더 높은 BE를 보여주고 있다. 따라서 채널 오류율에 따라서 적합한 전송 오류 복구 방안을 취사선택할 필요가 있다고 판단된다.

참고문헌

- [1] S. Deering, R. Hidden, "Internet Protocol, Version6 (IPv6) Specification", IETF RFC 2460, Dec. 1998.
- [2] V. Jacobson, "Compressing TCP/IP Headers for Low-Speed Serial Link", IERF RFC 1144, Feb. 1990.
- [3] M. Degermark, B. Nordgerm and S. Pink, "IP Header Compression", IETF RFC 2507, Feb. 1999.
- [4] C. Borman et. al., "Robust Header Compression (ROHC)", IETF RFC 3095, July. 2001.
- [5] S. Casner and V. Jacobson, "IP/UDP/RTP Headers for Low-Speed Serial Links", IETF RFC 2508, Feb. 1999.
- [6] A. Giovanardi, G. Mazzini, M. Rossi and M. Zorzi, "Improved Header Compression for TCP/IP Over Wireless Links", Electronics letters, vol.36, no.23, pp.1953-1959, Nov. 2000.
- [7] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport Protocol for real time applications", IETF RFC 1889, Jan. 1996.
- [8] W.S. Filippo, M.W. Ritter, R.J Friday and A. Srivastava, "A study of TCP Performance over wireless data networks", in Proc IEEE Int. conf VTC, vol.3,

- pp2265-2269, 2001.
- [9] A. Kondo, A. Sadka, S. Worrall, S. Febri and A. Cellatoglu, "Robust header compression for wireless networks", in Proc IEEE Int. conf. Local Computer Networks, pp.377-378, 2001.
- [10] F. Fitzek, S. Hendrata, P. Seeling, M. Reisslein, "Header Compression Schemes for Wireless Internet Access", Jul.2003.
- [11] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control(MAC) and Physical Layer (PHY) Specifications", in IEEE Std 802.11-1999, 1999.
- [12] G. Boggia, P. Camarda and V.G. Squeo, "ROHC+: a new header compression schemes for TCP streams in 3G wireless systems", in Proc IEEE Int. conf. Communications, vol.5, pp.3271-3278, 2002.
- [13] M. Degermark, L.-E. Jonsson, H. Hannu and K. Svanbro, "Wireless real-time IP services enabled by header compression", in Proc IEEE int. conf. VTC, vol.2, pp.1150-1154, 2000.
- [14] T. Koren, S. Casner, J. Geevarghese, "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering", IETF RFC 3545, July, 2003.
- [15] J. Haipeng, H. Raymond, W. Jun, "Performance Comparison of Header Compression Schemes for RTP/UDP/IP Packets", in Proc IEEE Int. conf. WCNC, pp.1691-1696, 2004.
- [16] Y. Alkiviadis, K. Edger, S.A. Alcatel, "Evaluation of Header Compression Schemes For IP-Based Wireless Access System", in IEEE Wireless Communications, pp.68-74, Feb, 2005.
- [17] F.H.P. Fitzek, T. K. Madsen, P. Popovski and R. Prasad, "Cooperative IP Header Compression for Parallel Channels in Wireless Meshed Networks", in Proc IEEE Int. conf. ICC, May 2005, Seoul, Korea.
- [18] Vijay Suryavanshi and A. Nosratinia, "Error resilient packet header compression," submitted to IEEE Trans. Commun., July 2005.
- [19] F.H.P. Fitzek and T.K. Madsen and R. Prasad and M. Katz, "Cooperative Header Compression for 4G Wireless Networks," Proc. of 13th Wireless World Research Forum (WWRF)}, March 2005, Jeju Island, Korea.
- [20] M. Degermark, M. Egan, B. Nordgen, and S. Pink, "Low-loss TCP/IP header compression for wireless networks," Wirel. Netw., vol.5, pp. 375-387, 1997.
- [21] S. J. Perkins, and M. W. Mutka "Dependency removal for transport protocol header compression over noisy channels," Proc. ICC'97.
- [22] A. Calvers-Aug, M. Arnau-Osorio, and J. Paradells-Aspas "An improvement of TCP/IP header compression algorithm for wireless links," Proc. ISAS'99.

저자소개



신 병 철 (Brian B. Shinn)

KAIST 전자공학과 석사
KAIST 전자공학과 박사
KAIST 교수
충북대학교 교수

※ 관심분야: 멀티미디어통신, 무선랜

김 민 영 (Min-Yeong Kim)

충북대학교 전과공학과 학사졸업
충북대학교 전과공학과 석사졸업

※ 관심분야: 광대역 네트워크, BcN, 휴대인터넷

홍고르출 (Khongorzul D.)

Bachelor of Mathematics in Nat. Univ. of Mongolia (NUM),
Mongolia

Master of Math. in NUM, Mongolia

Ph.D Student of Radio Eng. in Chungbuk National
University (CBNU), Korea, 2005.3 - present

※ Interest: Mobile/Multimedia Communication Syst

이 인 성 (In-Sung Lee)

한국통신 연구 개발단 전임연구원 Texas A&M Univ..
Dept. of Electrical Eng.

ETRI 이동통신기술연구단 연구원

충북대학교 부교수

※ 관심분야: 이동/위성통신 시스템, 신호처리