

일반논문-11-16-3-14

IP 망을 통한 비디오 전송에 효율적인 망 적응적 ARQ 오류제어 기법

심 상 우^{a)}, 서 광 덕^{b)*}, 김 진 수^{c)}, 김 재 곤^{d)}, 정 순 흥^{e)}, 배 성 준^{e)}

Network Adaptive ARQ Error Control Scheme for Effective Video Transport over IP Networks

Sang-woo Shim^{a)}, Kwang-deok Seo^{b)}, Jin-soo Kim^{c)}, Jae-Gon Kim^{d)}, Soon-heung Jung^{e)}, and Seong-Jun Bae^{e)}

요 약

본 논문에서는 패킷 오류가 발생하는 IP 망을 통해 비디오 전송 서비스를 제공하기 위한 효율적인 망 적응적 ARQ 기반의 오류 제어 기법을 제안한다. 재전송에 요구되는 지연시간이 허용되고 재전송을 위한 피드백 채널의 사용이 가능한 경우, IP 망에서 패킷 손실이 발생하면 수신 단말은 서버에게 손실된 패킷의 재전송을 요청할 수 있다. 그러나 망 상황과 동시에 전송되는 피드백 메시지의 수를 고려하지 않고 무조건적인 재전송을 요청할 경우 손실된 패킷이 복호화에 필요한 시점까지 도착할 것이냐가 보장되지 않는다. 제안된 ARQ 기법은 추정된 망 상태를 기반으로 조건적으로 재전송 요청 여부를 결정하고, 유효한 재전송 요청 시점을 결정하여 보다 효율적인 ARQ가 적용될 수 있도록 한다. 제안된 망 적응적 ARQ 기반의 오류 제어 기법의 성능을 검증하기 위하여, 패킷 손실 네트워크 환경을 NIST-Net을 기반으로 구현하며 실험 결과를 통해 제안된 기법의 우수한 오류 강인 성능을 검증한다.

Abstract

In this paper, we propose an effective network-adaptive ARQ based error control scheme to provide video streaming services through IP networks where packet error usually occurs. If time delay and feedback channel are allowed, client can request server to retransmit lost packets through IP networks. However, if retransmission is unconditionally requested without considering network condition and number of simultaneous feedback messages, retransmitted packets may not arrive in a timely manner so that decoding may not occur. In the proposed ARQ, a client conditionally requests retransmission based on assumed network condition, and it further determines valid retransmission time so that effective ARQ can be applied. In order to verify the performance of the proposed adaptive ARQ based error control, NIST-Net is used to emulate packet-loss network environment. It is shown by simulations that the proposed scheme provides noticeable error resilience with significantly reduced traffics required for ARQ.

Keywords : ARQ, packet error control, video packet retransmission, video transport over IP networks.

- a) 안철수연구소
AhnLab
- b) 연세대학교
Yonsei University
- c) 한밭대학교
Hanbat National Univeristy
- d) 한국항공대학교
Korea Aerospace University
- e) 한국전자통신연구원
ETRI

* 교신저자 : 서광덕(kdseo@yonsei.ac.kr)

※ 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2010-0025268).
· 접수일(2011년3월25일), 수정일(2011년5월9일), 게재확정일(2011년5월9일)

1. 서론

최근 통신 기술이 발전함에 따라 통신망의 속도도 빠르게 증가해 왔다. 유선 통신의 경우 FTTH (fiber to the home) 기술을 기반으로 각 가정에 초고속 인터넷이 공급되고, 무선 통신의 경우 현재 제공되고 있는 3G 이동통신보다 더욱 빠른 통신 속도를 제공하는 4G 이동통신으로 발전할 예정이다. 이러한 가입자 접속 네트워크의 광대역화로 인해 인터넷 접속 속도가 비약적으로 증가함에 따라 고품질 멀티미디어 서비스에 대한 욕구도 증가해 왔다. 특히 IP 인프라를 기반으로 한 유무선 스트리밍 및 IPTV 서비스의 확대는 IP 망에서의 고품질 비디오 서비스 제공을 위한 진보된 기술 개발의 중요성을 촉진시키고 있다.

유선의 경우 최근 패킷 네트워크에 기반을 둔 IPTV 서비스가 상용화 되어 대중화에 성공했으며, 무선의 경우 DMB 및 영상통신 서비스에서 더 나아가 무선 IPTV를 상용화하기 위한 노력이 계속 되어왔다. 현재 인터넷을 통한 실시간 비디오 전송 서비스에서 실질적으로 미디어 전송을 담당하는 프로토콜은 RTP (Real-time Transport Protocol)이다^{[1][2]}. IP 기반의 비디오 전송에 적합하도록 데이터량을 감축시키는 비디오 압축 과정을 통해 생성된 비트스트림은 RTP로 패킷화되어 IP 망으로 전달되고 RTCP (RTP Control Protocol)에 의해 전송된 패킷의 수신에 관련된 품질을 추정할 수 있다^[2].

통신망의 속도가 빨라지고 고품질 멀티미디어 서비스에 대한 욕구는 증가해 왔지만, 서비스의 기반이 되는 IP 기반의 패킷 네트워크는 안정적인 멀티미디어 서비스를 제공하기 위한 QoS를 제공하지 않는다. 즉, 패킷 전송이 불안정하거나 패킷 손실, 전송 지연, 채널 대역폭 등의 변동이 발생할 가능성이 높고, 패킷 오류를 줄이거나 제어하기 위한 추가적인 기능을 기본적으로 제공하지 않는다. 그러나 고품질 멀티미디어 서비스의 사용자들은 안정적인 영상서비스를 제공받길 원한다. 따라서 사용자들의 QoS를 만족시키면서 패킷 오류를 극복할 수 있는 기법이 필요하다. 그 중에서도 무선 채널의 페이딩 현상과 채널 잡음으로 인한 다량의 채널 오류가 원인이 되는 패킷 손실은 IP 망에서 근본적으로 발생하는 문제점이다^[3].

이러한 채널의 패킷 손실을 극복하기 위해서 일반적인 무선 네트워크는 링크계층 (link layer)에서 ARQ (Automatic Repeat Request)와 같은 전송 오류 제어를 수행할 수 있다^[4]. 그러나 이러한 접근 방법을 통해 모든 형태의 패킷 오류를 극복할 수 없으며, 상위 계층에서 패킷 손실 및 전송 지연의 변동을 유발할 수 있고 스트리밍 비디오의 급격한 품질의 열화를 야기시킬 수 있다. 따라서 채널의 패킷 오류 형태 변동이 심한 유무선 인터넷 환경에서 성공적으로 비디오를 전송하기 위해서 응용계층 (application layer)에서 네트워크에 적응적인 전송오류 제어 기법의 개발이 필요하다. 응용 계층의 대표적인 오류 제어 방식은 ARQ와 FEC (Forward Error Correction)로 나눌 수 있다^[4]. ARQ와 FEC 중 어느 것이 더 효율적인지는 사용되는 시스템의 특성, 경제성, 장비와 단말의 기능, 사용자의 요구 사항 등에 따라 달라질 수 있다. 그러나 서비스 환경에서 ARQ에 필요한 피드백 채널이 지원된다면 ARQ가 FEC보다 유리한 측면이 많다^{[5][6]}. 기본적으로 FEC는 첨가되는 잉여패킷 (parity packet)들로 인해 채널의 대역폭을 낭비시켜 시스템의 효율을 저하시키지만, 재전송에 필요한 버퍼와 그에 따른 지연시간을 고려하지 않아도 되기 때문에 멀티캐스트와 같은 실시간 전송에서 발생하는 패킷 오류에 효율적으로 대처할 수 있다^{[6][7]}. 또한, RTP와 RTCP를 위한 별도의 포트 (port) 설정을 통해 RTCP 기반의 피드백 채널 활용이 가능한 IP 망에서는 유니캐스트 (unicast)와 멀티캐스트 (multicast) 서비스의 경우 RTCP 패킷을 응용하여 손실된 패킷에 대한 NACK 메시지를 전송할 수 있기 때문에 ARQ를 통해 패킷 오류에 효율적으로 대처할 수 있다^[8].

본 논문에서는 단말에서 NACK 피드백을 위한 RTCP 대역폭으로 활용 가능한 채널 대역폭을 바탕으로 현재 발생하는 패킷 오류를 효율적으로 제어하기 위한 망 적응적 ARQ 기법을 제안한다. 논문에서 제안된 ARQ 기법에서는 손실된 패킷에 대한 효율적인 ARQ를 수행하기 위해서, 재전송된 패킷이 유효한 시간 내에 도착 가능한지의 가능성을 바탕으로 재전송 여부를 판단하고 재전송 요청 여부를 결정한다. RTCP 피드백을 위한 가용 대역폭에 따라 패킷 손실 시 재전송 요청을 즉시 요청할 것인지, 또는 손실된 여러 개의 패킷에 대한 재전송 요청을 가용 대역폭에 여유가 있을 때 한꺼번에 모아서 재전송을 요청할 것인지 등을 결정하게 된다.

II. 관련 연구

ARQ는 수신측이 송신측에게 손실된 패킷에 대해 재송신해 줄 것을 요구하는 에러 통제 기법이다. ARQ 프로토콜은 일반적으로 다음과 같이 동작한다. 패킷 손실이 발생하면 수신측은 송신측에 패킷 재전송을 요구하고, 재전송을 요구받은 송신측은 손실이 발생한 패킷을 재전송한다. ARQ는 손실된 패킷에만 부가적인 오류 제어를 실시하기 때문에 효율적이지만 패킷이 손실된 것을 감지하는 시간과 패킷의 재전송을 요구하는데 걸리는 시간, 또 손실된 패킷을 송신측에서 다시 전송하는데 걸리는 시간을 필요로 한다. 따라서 ARQ는 전송 지연 시간이 작은 환경에서 사용하면 효과적으로 전송 에러를 복원할 수 있다^{[3][5][6][9]}.

ARQ는 송신측과 수신측에서 불필요한 패킷 전송을 방지하기 위하여 1) 재전송 요구를 수신측에서 제어하는 방식 (receiver-based ARQ)과, 2) 패킷 재전송 여부를 송신측에서 제어하는 방식 (sender-based ARQ)을 활용하여 재전송을 제어할 수 있다.

1. 수신측 기반 ARQ

수신측 기반 ARQ는 수신측에서 재전송 패킷이 유효한

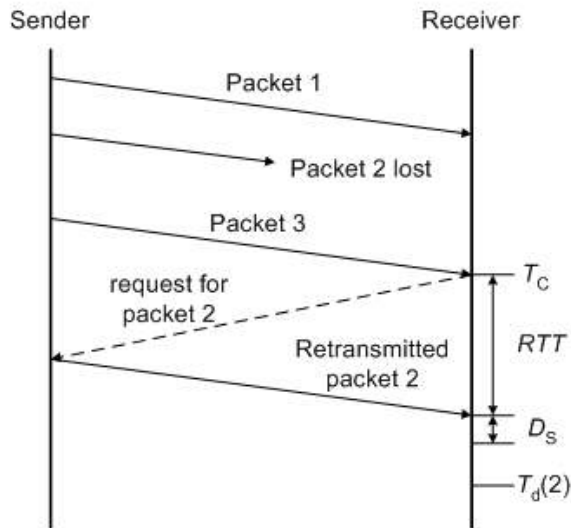


그림 1. 수신측 기반 ARQ
Fig. 1. Receiver-based ARQ

시간에 맞게 도착하는지 여부를 판단하여 재전송 패킷이 시간에 맞게 도착하지 못하는 재전송 시도를 사전에 차단한다. 그림 1은 수신측 기반 ARQ 방식의 동작 원리를 나타낸다.

수신측에서 재전송을 요청할지 여부를 판단하는 기준은 다음 조건과 같다.

수신측이 패킷 N 이 손실되었음을 발견했을 때:
 $if(T_c + RTT + D_s < T_d(N)) \rightarrow$ 송신측에 패킷 N에 대한 재전송을 요구한다.

상기의 조건에서 T_c 는 현재 시간, RTT 는 왕복지연시간 (round-trip time)에 대한 추정치, D_s 는 수신측에서 소요되는 대략적인 처리 시간, $T_d(N)$ 은 패킷이 문제없이 디코딩 되기 위해 도착해야 할 한계 시간 (deadline)을 의미한다.

2. 송신측 기반 ARQ

송신측 기반 ARQ는 송신측에서 재전송 패킷이 유효한 시간에 맞게 도착하는지 여부를 판단하여, 재전송 패킷이 시간에 맞게 도착하지 못하는 재전송 시도를 사전에 차단한다. 그림 2는 송신측 기반 ARQ 방식의 동작 원리를 나타낸다.

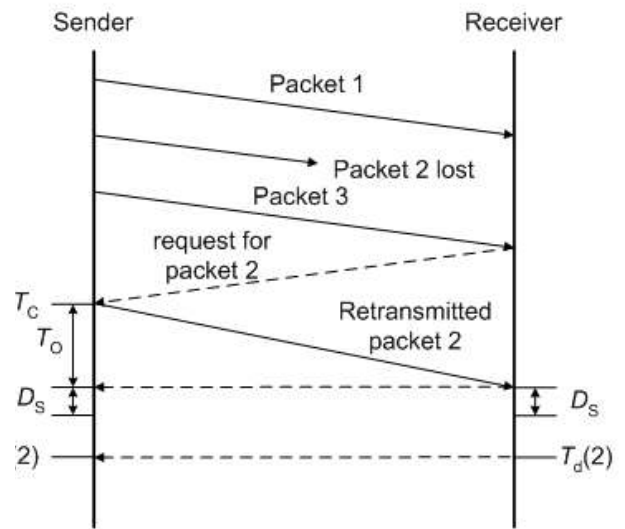


그림 2. 송신측 기반 ARQ
Fig. 2. Sender-based ARQ

송신측에서 재전송을 실시할지 여부를 판단하는 기준은 다음 조건과 같다.

송신측이 패킷 N 에 대한 재전송 요청을 받았을 때:
 $if(T_c + T_o + D_s < T_d'(N)) \rightarrow$ 송신측이 패킷 N을 수신 측에 재전송한다.

이 식에서 T_o 는 송신측에서 수신측으로의 종단간 지연의 추정치, $T_d'(N)$ 은 송신측에서 추정된 $T_d(N)$ 을 나타낸다. $T_d(N)$ 을 얻기 위해서는 수신측에서 송신측으로 관련 정보를 전달해야 하는데, RTCP 수신보고 (receiver report) 패킷이 이러한 정보를 전달하는 역할로 활용될 수 있다.

III. 제안된 망 적응적 ARQ 기법

ARQ는 손실된 특정 패킷에 대해서만 재전송 요청을 통한 오류 제어를 하기 때문에 효율적이지만 패킷이 손실된 것을 감지하는 시간과 패킷의 재전송을 요구하는데 걸리는

시간, 그리고 손실된 패킷을 보고 받은 송신측에서 다시 재전송하는데 걸리는 시간을 추가적으로 요구한다. 즉, ARQ 기법은 손실된 패킷만을 특정하여 송신측에서 재전송하는 방식으로 오류 복구에 대한 신뢰성 측면에서 효율적이긴 하지만 재전송에 소요되는 시간에 의해 예측하기 쉽지 않은 지연을 야기시키므로, 엄격한 실시간의 제약을 갖는 멀티미디어 응용서비스에서는 미디어 패킷의 유효 시간을 고려하여 적용되어야 한다. 따라서, 전송 지연이 작은 서비스 환경이나 비디오 서비스의 초기 버퍼링(initial buffering)을 통해서 지연과 신뢰성 사이의 상충적인 관계 (trade-off)를 적절히 고려한다면 ARQ를 통해 손실된 패킷을 효율적으로 복원시킬 수 있다^[8]. 제안하는 기법에서는 유효하지 못한 패킷에 대한 재전송이 이루어질 것으로 판단되는 경우에 대해서는 사전에 재전송 자체를 요청하지 않으므로써 재전송 요청에 소요되는 불필요한 대역폭의 낭비까지도 회피할 수 있는 장점을 지닌 그림 1에 나타난 수신측 ARQ 제어 방식을 적용한다.

제안하는 망 적응적 ARQ 알고리즘은 패킷 손실이 검출되었을 때 피드백 채널의 대역폭 상황을 고려하여 즉각적으로 손실된 패킷에 대한 재전송을 요청하는 Immediate 모

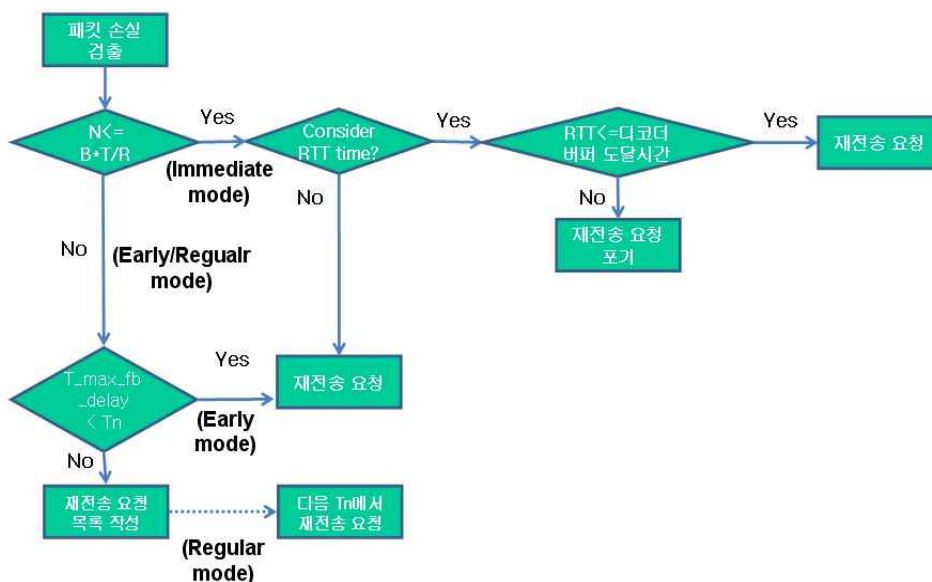


그림 3. 제안하는 망 적응적 ARQ 기법의 블록도
 Fig. 3. Block diagram of the proposed network-adaptive ARQ scheme

드와 재전송 시점을 일정 시간 동안 지연시키는 Early 모드 및 Regular 모드로 분류된다. Immediate 모드는 주어진 피드백 채널의 망 상태가 즉각적으로 재전송 요청 메시지를 발송할 수 있을 정도로 채널의 대역폭 상태가 양호한 경우에 선택된다. Early 모드는 피드백 채널의 망 상태가 즉각적으로 재전송 요청 메시지를 발송할 수 없는 상황에서 일정 시간 동안의 허용 가능한 지연 이후에 재전송 요청 메시지를 발송하는 경우이다. Regular 모드는 피드백 채널의 망 상태가 즉각적으로 재전송 요청 메시지를 발송할 수 없는 상황에서 다음에 발송이 예정된 정규 RTCP RR (receiver report) 패킷 발송 때 재전송 요청 메시지를 함께 동봉하여 재전송을 요청하게 되는 경우이다. 이러한 3가지 모드를 포함하는 제안하는 ARQ 기법의 블록도는 그림 3에 나타나 있다.

그림 3에 제시된 제안하는 ARQ 기법은 크게 두 가지 단계로 이루어져 있다. 첫 번째 단계에서 현재의 망 상황을 고려하여 Immediate 모드와 Early/Regular 모드 가운데 적합한 모드를 선택한다. 두 번째 단계에서는 첫 번째 단계에서 Immediate 모드로 선택된 경우 RTT를 고려하여 재전송 요청의 여부를 최종적으로 결정하는 절차가 포함되어 있고, 첫 번째 단계에서 Early/Regular 모드로 선택된 경우 RTT를 고려하여 Early 모드와 Regular 모드 가운데 하나를 선택하는 절차가 포함되어 있다.

첫 번째 단계에서 Immediate 모드와 Early/Regular 모드 가운데 적합한 모드를 선택하기 위해서는 그림 3에 나타나 있는 4가지 파라미터를 고려한다. 이 4가지 파라미터에는 피드백 패킷 전송 간격 (패킷 손실 간격에 해당함) 시간의 평균적인 크기를 나타내는 T (sec), T 시간 간격 동안 수신측에서 보고하게 되는 피드백의 평균적인 회수를 나타내는 N , 피드백 메시지 전송을 위하여 수신측에서 가용한 대역폭인 B (bytes/sec), 그리고 평균적인 피드백 패킷의 크기를 나타내는 R (bytes)이 포함된다. 한편, RFC3550 문서를 기반으로 구현된 RTCP 피드백 채널은 주기적인 송신자보고 (Sender Report)와 수신자보고 (Receiver Report)를 통해 RTP 세션의 채널 품질에 관련된 정보를 측정할 수 있다 [2]. 또한, RTCP 전송을 위한 피드백 채널의 대역폭은 RTP 채널 대역폭의 5%를 사용하도록 권고되고 있다^[2]. 특정 시

점에서 전송이 필요한 재전송 요청을 위한 피드백 메시지를 즉각적으로 발송하기 위해서는 다음과 같은 관계식이 만족되어야 한다.

$$N \leq B \times T / R \quad (1)$$

그림 3에 보이듯이, 식 (1)을 만족할 경우에는 현재 전송이 필요한 재전송 요청 메시지의 발송이 즉각적으로 가능한 상황으로써 Immediate 모드로 선택된다. 만약 식 (1)을 만족하지 않는 경우에는 즉각적인 재전송 요청 메시지의 발송을 보류하고 Early 모드와 Regular 모드 가운데 하나를 선택하고 두 번째 단계로 넘어가게 된다.

식 (1)을 통해 Immediate 모드로 선택된 경우 채널의 RTT를 고려하여 재전송 요청 여부를 최종 결정한다. 이때, 버퍼에 남아 있는 패킷 데이터 량이 시간적으로 RTT보다 큰 경우에는 재전송 요청이 유효하다고 판단하여 재전송을 요청하게 되며, 그렇지 않을 경우에는 재전송 요청을 포기하게 된다. 식 (1)을 통해 Early 모드 또는 Regular 모드로 선택된 경우에는 버퍼에 잔존하는 패킷의 시간적인 량과 정규 RTCP 패킷의 전송 시점을 비교하여 Early 모드 또는 Regular 모드 가운데 하나를 선택하게 된다. 그림 4는 Early 모드와 Regular 모드를 결정하기 위해 고려되는 타이밍 모델 (timing model)을 나타낸다. 그림에서 t_0 는 패킷 손실이 감지되었을 때 재전송 모드를 결정하는 시점을 나타내며, t_p 는 직전에 전송된 RTCP 패킷의 전송 시점, t_n 은 다음에 예정된 정규 RTCP 패킷의 전송 시점을 나타낸다. t_0 에서 식 (1)을 만족하지 않으면 $T_{max_fb_delay}$ 를 계산하여 t_0 로부터 $T_{max_fb_delay}$ 이후의 시점과 다음에 예정된 정규 RTCP 전송 시점인 t_n 를 비교한다. $T_{max_fb_delay}$ 는 재전송된 패킷이 RTT 시간만큼 소요되어 수신 단말의 버퍼에 도착한 이후에도 버퍼에 남아 있는 패킷으로 재생 서비스가 가능한 최대의 시간을 의미한다. 즉, $T_{max_fb_delay}$ 는 손실 패킷에 대한 재전송 요청이 유효하기까지 허용이 가능한 최대의 지연 시간을 의미하며 다음과 같이 계산된다.

$$T_{max_fb_delay} = T_d - RTT \quad (2)$$

식 (2)에서 T_d 는 수신 버퍼에 저장된 마지막 패킷이 버퍼에서 비워질 때까지 소요되는 남아 있는 시간을 의미한다. T_d 와 RTT 간의 차이를 통해 재전송 요청을 지연시킬 수 있는 최대의 한계 시간을 구할 수 있는데, 이 시간이 t_n 에 도달하기 까지 남아 있는 시간보다 작으면 (즉, $T_{max_fb_delay} < t_n$), $T_{max_fb_delay}$ 이내의 구간 내에서 Early 모드 재전송 요청을 t_e 시점에서 하게 된다. 반대로 t_n 에 도달하기 까지 남아 있는 시간보다 길면 (즉, $T_{max_fb_delay} > t_n$), t_n 까지 기다려서 정규 RTCP 패킷이 전송될 때 손실된 패킷에 대한 재전송 요청을 함께 포함시키게 된다.

한편, $T_{max_fb_delay}$ 값이 음수가 될 때는 현재의 RTT가 너무 큰 관계로 재전송 요청을 한 패킷이 유효한 시점을 초과하여 도착하게 되는 경우이다. 이러한 경우에는 무의미한 재전송 요청으로 인해 망의 트래픽을 가중시키는 것을 방지하기 위해서 재전송 요청을 포기하게 된다. 그림 3의 제안된 ARQ 기법에서는 RTT가 중요하게 활용되고 있

는데, RTT 추정을 위한 RTCP 피드백 패킷을 활용한 일반적인 RTT 추정 방법이 그림 5에 제시되어 있다.

서버에서 SR 패킷을 송신하면 D1의 지연 이후에 클라이언트에 도착하며, 클라이언트는 그에 대한 응답으로 RR 패킷을 서버에게 전송하며 서버에 도착하기 까지 D2의 지연이 존재한다. 이때, 클라이언트가 SR 패킷을 수신한 다음 RR 패킷을 보내기까지 준비에 소요되는 지연인 DLSR (delay since last SR)이 존재하므로 RTT는 다음 식과 같이 계산할 수 있다.

$$RTT = D1 + D2 = A - LSR - DLSR \quad (3)$$

식 (3)에서 LSR (last SR)은 서버가 SR 패킷을 전송한 시점의 절대시간 (NTP 타임스탬프로 표현)을 의미하며 A는 RR 패킷이 서버에 도착했을 때의 절대시간을 의미한다. 그런데, 이 방법은 RTT를 송신측에서 추정한다는 측면에

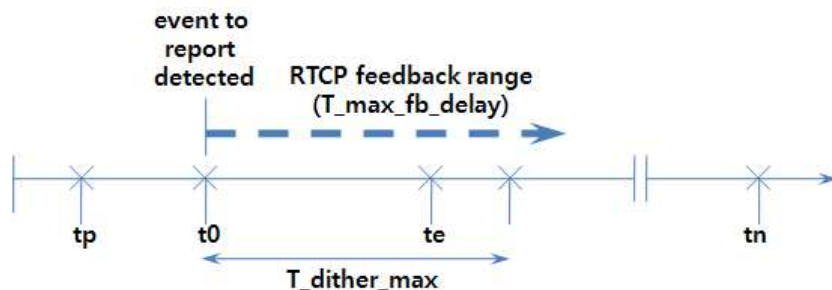


그림 4. 재전송 요청 모드 결정을 위한 타이밍 모델
Fig. 4. Timing model for determining retransmission-request mode

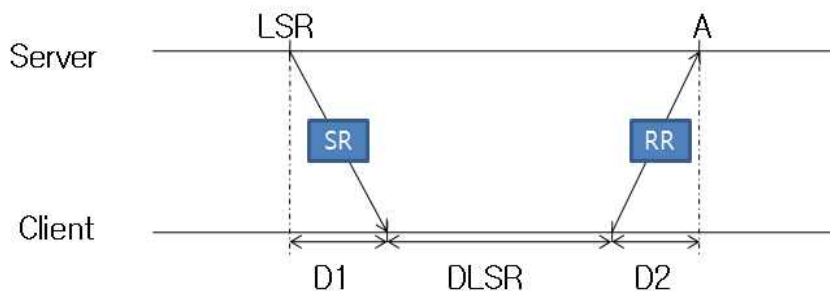


그림 5. RTCP 패킷을 활용한 일반적인 RTT 추정 방법
Fig. 5. General RTT estimation method using RTCP packet

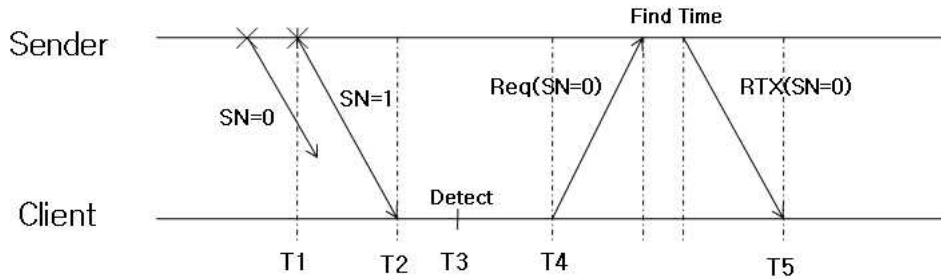


그림 6. 제안하는 수신측에서의 RTT 추정 기법
 Fig. 6. Proposed RTT estimation method at the receiver side

서 그림 2의 송신측 기반 ARQ에서 효율적으로 활용이 가능하며, 본 논문에서 제안하는 수신측 기반 ARQ를 위해서는 새로운 RTT 추정 기법이 필요하다. 그림 6은 수신 측에서 RTT를 측정하는 실용적인 방법의 예를 나타낸다.

그림 6에서는 순서번호 (sequence number) 0번째 패킷 (SN=0)이 손실되어 이 패킷에 대한 재전송 요청을 절대시간 T4에서 수신측에서 하게 되고, 서버가 재전송 요청 메시지를 수신 후에 재전송 패킷을 검색하여 재전송을 하게 되면 수신측에서는 T5 시점에 재전송된 패킷을 수신하게 된다. 이때 수신측에서 측정하게 되는 RTT는 재전송 요청 시점부터 재전송 패킷을 수신하기까지의 총 지연 시간에 해당되며 다음 식과 같이 계산된다.

$$RTT = T5 - T4 - Find\ Time \quad (4)$$

식 (4)에서 Find_Time은 서버가 재전송 요청된 패킷을 전송 버퍼에서 검색하여 재전송을 준비하는데 까지 걸리는 시간을 의미한다. 이러한 수신측 기반의 RTT 추정 기법은 제안하는 수신측 기반의 ARQ 기법에 보다 효과적으로 적용이 가능하고, 패킷 재전송 과정을 통해 실측된 RTT를 활용하므로 패킷 재전송을 목적으로 하는 ARQ에 보다 더 정확한 RTT 값을 제공할 수 있다. 본 논문에서는 망 내부에서의 순간적인 일시적 변동에 영향을 받아 측정된 RTT 값에 의한 과도한 영향을 조절하기 위해서 다음과 같은 이동 평균 (moving average)을 통해서 RTT 값을 업데이트 한다.

$$RTT(n+1) = (1-\alpha) \times RTT(n) + \alpha \times RTT \quad (5)$$

식 (5)에서는 기존에 얻어진 $RTT(n)$ 값에 현재 측정된 RTT 값의 영향력을 가중치 α 값 ($0 \leq \alpha \leq 1$)을 통하여 적절히 반영시켜서 업데이트된 $RTT(n+1)$ 값을 얻어낸다.

IV. 실험 결과

실제적인 네트워크 환경에서 제안된 방법의 성능을 평가하기 위하여 IP 네트워크 에뮬레이터인 NIST-Net을 활용하여 실험을 하였다^[10]. NIST-Net 패키지를 이용할 경우 원하는 패킷 손실 확률을 입력하여 채널에서의 패킷 손실 강도를 조절할 수 있다. 또한 일정한 대역폭을 가지는 가상 네트워크를 생성할 수 있고 가상 네트워크에서 사용되는 라우터는 DRD (derivative random drop)^[11] 메커니즘을 기

표 1. NIST-Net 기반의 전송 채널에서의 평균 패킷 손실 확률과 패킷 전송 지연

Table 1. Average packet-loss ratio and packet-transmission delay in the NIST-Net based transmission channel

평균 패킷 손실 확률	0 %	1 %	2 %	3 %
평균 패킷 전송 지연	0 ms	0 ms	0 ms	0 ms
	100 ms	100 ms	100 ms	100 ms
	200 ms	200 ms	200 ms	200 ms
	300 ms	300 ms	300 ms	300 ms
	400 ms	400 ms	400 ms	400 ms
	500 ms	500 ms	500 ms	500 ms

반으로 채널에서의 패킷 손실을 발생시키고 패킷 손실률을 조절할 수 있다. 본 실험에서는 NIST-Net에 의해 구성된 가상 IP 네트워크를 통하여 H.264^[12] 비디오를 전송하면서 패킷 오류 제어와 관련된 다양한 시뮬레이션을 수행하였다. 실험에 적용될 전송 채널에서의 패킷 손실 환경은 표 1과 같이 설정되었다. 표 1에서는 평균 패킷 손실 확률과 전송 지연 시간을 나타낸다.

구현된 시스템의 성능을 검증하기 위하여 재전송 요청 모드 결정 기법 사용 여부에 따라 최종 손실 패킷 수, 전체 패킷 대비 최종 손실 패킷 비율 등을 측정하였다. 각각의

경우에 적용된 패킷의 크기는 인터넷의 MTU (maximum transfer unit) 크기가 1500 바이트임을 고려하여 모두 1000 바이트로 동일하게 유지하였다. 실험에서는 수신측 버퍼에서 초기 버퍼링 시간을 서비스 사용자의 지연시간에 대한 민감도와 서비스의 안정성을 동시에 고려하여 일반적인 상용서비스에서 적용되는 7초 정도로 설정하여서 사용자의 서비스 지연 시간에 대한 만족도를 유지함과 동시에 비디오가 최대한 끊김 없이 동작할 수 있도록 하였다.

수신 버퍼의 크기가 작으면 디코딩을 위해서 버퍼에서 데이터를 끄집어 내는데 걸리는 시간이 짧아지므로 영상

표 2. 수신측 초기 버퍼링 시간을 7초로 설정한 경우의 실험 결과 (NIST-Net에서의 채널 패킷 손실률이 각각 1%, 2%, 3% 일 때)
Table 2. Experimental results for the initial buffering time of 7 seconds at the receiver (for 1%, 2%, and 3% packet-loss ratio)

1%							
구분	망 적응적 오류제어 모듈 사용				망 적응적 오류제어 모듈 미사용		
	재전송 요청포기 비율 (%)	최종손실 패킷 수	최종 패킷손실률 (%)	절약된 피드백 대역폭비율(%)	재전송 패킷 수신실패	최종손실 패킷 수	최종 패킷 손실률 (%)
0ms	8.0	0	0	8.0	0	0	0
100ms	8.2	2	0.0022	8.2	1	1	0.0013
200ms	8.5	16	0.0212	8.5	6.7	14	0.0186
300ms	10.0	34	0.0446	10.0	15.7	26.3	0.0349
400ms	11.4	112	0.1479	11.4	34.7	89.3	0.1184
500ms	12.1	324	0.4290	12.1	107	282.7	0.3757

2%							
구분	망 적응적 오류제어 모듈 사용				망 적응적 오류제어 모듈 미사용		
	재전송 요청포기 비율 (%)	최종손실 패킷 수	최종 패킷손실률 (%)	절약된 피드백 대역폭비율(%)	재전송 패킷 수신실패	최종손실 패킷 수	최종 패킷 손실률 (%)
0ms	11.9	0	0.0004	11.9	4	4	0.0053
100ms	12.9	5	0.0062	12.9	4.3	4.7	0.0062
200ms	13.4	34	0.0455	13.4	22.7	34	0.045
300ms	14.9	83	0.11	14.9	35.3	87	0.1153
400ms	15.0	206	0.2741	15.0	43	192	0.2549
500ms	15.6	567	0.7574	15.6	88.3	514	0.6853

3%							
구분	망 적응적 오류제어 모듈 사용				망 적응적 오류제어 모듈 미사용		
	재전송 요청포기 비율 (%)	최종손실 패킷 수	최종 패킷손실률 (%)	절약된 피드백 대역폭비율(%)	재전송 패킷 수신실패	최종손실 패킷 수	최종 패킷 손실률 (%)
0ms	12.4	2	0.0031	12.4	2.7	2.7	0.0035
100ms	13.3	9	0.0115	13.3	6.7	7.7	0.0102
200ms	14.4	75	0.0997	14.4	43	59.3	0.0786
300ms	14.8	149	0.1982	14.8	58.7	138.7	0.1840
400ms	15.1	359	0.4758	15.1	45.7	294.7	0.3917
500ms	15.9	893	1.1829	15.9	72	786.7	1.0527

서비스의 초기 대기 시간이 짧아지는 장점이 있다. 그러나 작은 버퍼 크기로 인해 안정적인 송수신이 어려워진다. 반대로, 수신 버퍼의 크기가 크면 안정적인 데이터 송수신이 가능하다. 즉, 버퍼의 크기와 영상서비스의 대기 시간 사이에는 상충 되는 (trade-off) 관계가 있다. 패킷 손실이 증가하면 재전송 요청이 증가한다. 그러나 짧은 시간에 많은 량의 재전송 요청이 발생하면 망에 부담을 줄 수 있다. 따라서 Early/Regular 모드와 같은 방법을 적용하게 될 경우 즉각적인 재전송 요청을 하지 않고, 망 상황을 고려하여 재전송 요청을 하도록 판단 할 수 있다. 표 2는 제안된 기법을 적용한 경우와 적용하지 않은 경우에 대한 실험 결과를 비교한다. 재전송 요청 포기 비율은 패킷 손실이 감지되어 재전송을 요청해야 하는 경우이지만 제안된 오류제어 기법을 적용하여 유효하지 못한 재전송 요청을 사전에 포기하는 비율을 나타낸다. 최종손실 패킷 수는 오류 제어 기법을 적용해서 최대한 복구를 했음에도 불구하고 복구를 하지 못한 패킷의 수를 나타낸다. 절약된 피드백 대역폭의 비율은 제안된 기법을 적용할 경우 유효하지 못한 재전송 요청을 미리 감지하여 재전송 요청을 하지 않음으로써 얻을 수 있는 피드백 대역폭의 절약 비율이다. 패킷 손실률이 1%일 경우에 제안된 기법에 의한 재전송 요청 포기 비율은 8~12%

가량 되며, 이로 인한 피드백 대역폭의 절약 비율도 8~12% 정도 된다. 패킷 손실률이 증가하여 3%인 경우에는 제안된 기법에 의한 재전송 요청 포기 비율이 다소 증가하여 12~15% 에 이르며, 이로 인한 피드백 대역폭의 절약 비율도 12~15% 에 이른다. 표에서 최종 패킷 손실률은 제안된 기법을 적용한 경우와 적용하지 않은 경우 사이에 큰 차이를 보이지 않는데, 그 이유는 기존의 기법의 경우 유효한 시점에 도착이 불가능한 패킷에 대해서도 무조건적으로 재전송을 요청하게 되는 반면에, 제안된 기법의 경우 유효한 시점에 도착이 불가능한 패킷에 대해서는 재전송 요청을 포기하기 때문이다. 즉, 제안된 기법의 경우 재전송을 요청하지 않았음에도 불구하고, 패킷 손실률 계산에는 재전송 요청의 포기로 인해 복구하지 못한 패킷도 포함되기 때문이다.

그림 7은 초기 버퍼링 시간을 7초로 설정한 경우의 재전송 요청 패킷 중에서 Early 및 Regular 모드에 의해 요청된 비율을 나타낸다. 1% 손실률에서 4~7%, 2% 손실률에서 11~14%, 그리고 3% 손실률에서 18~21%의 비율을 보인다. 이를 통해 전송 지연보다는 패킷 손실률에 의해 재전송 요청 모드 결정이 보다 더 큰 영향을 받는다는 것을 확인할 수 있다.

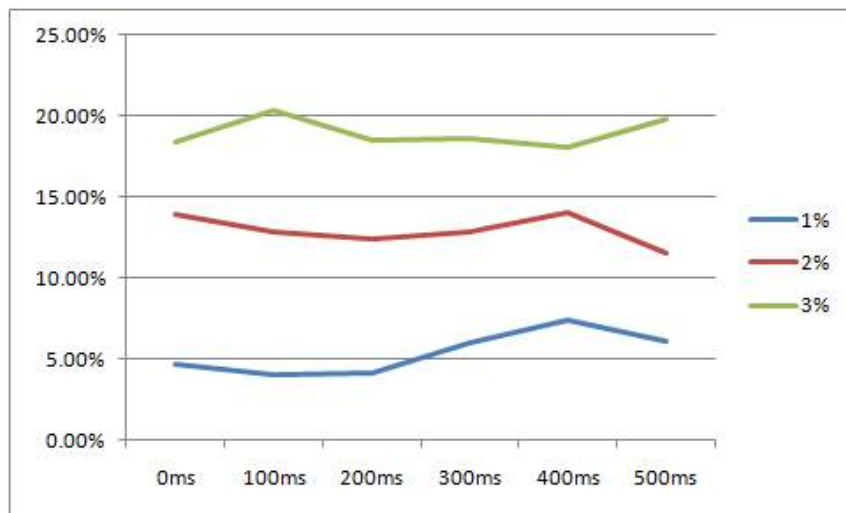


그림 7. 초기 버퍼링 시간을 7초로 설정한 경우의 Early/Regular 모드 요청 비율
 Fig. 7. Ratio of Early/Regular mode request for the initial buffering time of 7 seconds

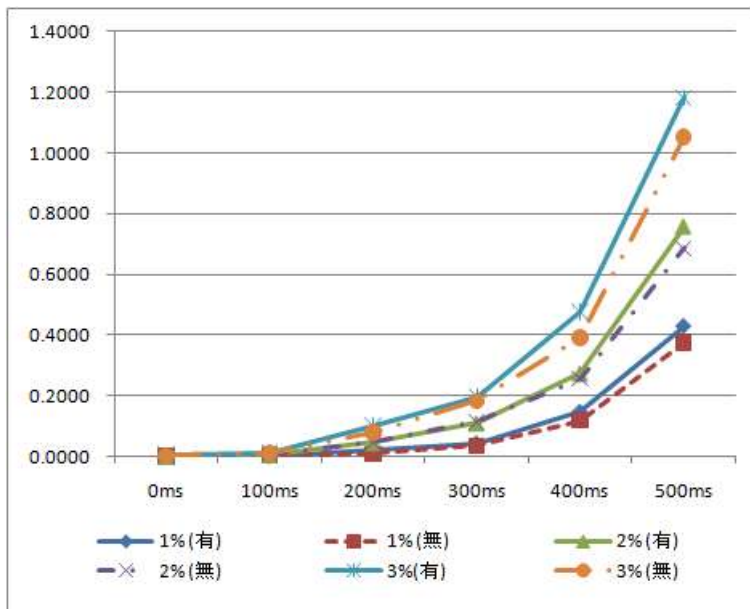


그림 8. 초기 버퍼링 시간을 7초로 설정한 경우의 최종 패킷 손실률(%)
 Fig. 8. Final packet loss ratio for the initial buffering time of 7 seconds

그림 8은 초기 버퍼링 시간을 7초로 설정한 경우에 전체 패킷(재전송 요청 패킷 포함) 중에서 최종 손실된 패킷의 비율을 나타낸다. 최종 패킷 손실의 계산에는 재전송 요청에도 불구하고 손실된 패킷, 재전송 되었으나 적절한 유효 도착 시점이 지나서 수신되어 버퍼에 저장되지 못한 패킷, 그리고 재전송이 되더라도 유효한 시점 이내에 도착하는 것이 불가능한 것으로 판단되어 재전송을 미리 포기한 패킷들을 모두 포함한다. 전체적으로 제안된 기법을 적용한 경우와 적용하지 않은 경우에 두드러진 성능상의 차이를 보이지 않는다. 그런데, 제안된 기법의 경우 유효한 시점 이내에 도착이 불가능한 패킷에 대한 재전송 요청을 포기함으로써 복구가 원천적으로 불가능하게 되는 패킷을 패킷 손실률 계산에서 제외할 경우 기존의 기법보다 훨씬 우수한 패킷 손실률을 나타내게 된다.

V. 결론

본 논문에서는 IP 망을 통한 비디오 전송에 효율적인 망 적응적 ARQ 기반의 오류제어 기법을 제안하였다. 제안된

기법에서는 IP 기반의 패킷 네트워크에서 패킷 손실, 패킷 지연, 망의 대역폭 변화 등이 발생했을 경우, 대응할 수 있는 재전송 요청 모드 결정 방법 및 재전송 요청 여부를 결정하는 알고리즘을 제안하였다. 제안된 망 적응적 ARQ 기법의 경우 망 상황에 적응적으로 대응하기 때문에 보다 효율적인 오류 제어가 가능하다. 실험 결과에서 현실적인 패킷 손실률 범위인 1~3% 범위에서의 패킷 손실률과 100~300ms의 전송 지연을 나타내는 IP 망 전송 환경에서 효율적인 오류 제어 성능을 확인할 수 있었다. 특히, 손실된 패킷에 대한 유효하지 못한 재전송 요청을 사전에 포기함으로써 대역폭의 낭비 요소를 미리 제거하여서, 망에 대한 오버헤드의 가중을 피하면서도 기존의 방법과 유사한 최종 패킷 손실률을 갖는 오류 제어 성능을 얻을 수 있었다.

참고 문헌

- [1] 김재균, "영상통신개론," 홍릉과학출판사, 2009년 8월.
- [2] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," IETF RFC 3550, IETF, July 2003.

- [3] H. Liu, H. Ma, M. Zarki, S. Gupta, "Error control schemes for networks: An Overview," *Mobile Networks and Applications*, vol. 2, no. 2, pp. 167-182, 1997.
- [4] F. Hartanto, H. Sirisena, "Hybrid error control mechanism for video transmission in the wireless IP networks," *IEEE Workshop on Local and Metropolitan Area Networks*, Sydney, Australia, Nov 1999.
- [5] F. Varcirca, A. Vendictis, A. Baiocchi, "Optimal design of hybrid FEC/ARQ schemes for TCP over wireless links with Rayleigh fading," *IEEE Trans. Mobile Computing*, vol. 5, no. 4, pp. 289-301, Apr 2006.
- [6] T. Gan, L. Gan, and K. Ma, "Expected run-time distortion based scheduling for scalable video transmission with hybrid FEC/ARQ error control," in *Proc, IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 1, pp. 829-832, Apr. 2007.
- [7] S. Aramvith, C. Lin, S. Roy, and M. Sun, "Wireless video transport using conditional retransmission and low-delay interleaving," *IEEE Trans. Multimedia*, vol. 12, no. 6, pp. 558-565, June 2002.
- [8] S. Wenger, M. Hannuksela, M. Westerlund, D. Singer, "RTP payload format for H.264 video," *IETF RFC 3984*, IETF, Feb. 2005.
- [9] A. R. Prasad, K. Seki, "Hybrid ARQ for IP Packet Transmission," *IEEE Universal Personal Communications Record*, vol 2, pp. 531-535, Oct 1997.
- [10] NIST-Net, Software provided by National Institute of Standards and Technology (NIST), <http://www-x.antd.nist.gov/nistnet>.
- [11] M. Gaynor, "Proactive packet dropping methods for TCP gateways," Available at <http://www.eecs.harvard.edu/~gaynor/final.ps>, Oct. 1996.
- [12] ISO/IEC JTC1/SC29/WG11, 14496-10 AVC, "Advanced video coding for generic audiovisual services," Mar. 2005.

저 자 소 개



심 상 우

- 2006년 2월 : 연세대학교 컴퓨터정보통신공학부 학사
- 2011년 2월 : 연세대학교 컴퓨터정보통신공학부 석사
- 2011년 2월 ~ 현재 : 안철수연구소 연구원
- 주관심분야 : 영상부호화, 영상통신, 정보보안



서 광 덕

- 1996년 2월 : KAIST 전기 및 전자공학과 학사
- 1998년 2월 : KAIST 전기 및 전자공학과 석사
- 2002년 8월 : KAIST 전기 및 전자공학과 박사
- 2002년 8월 ~ 2005년 2월 : LG전자 단말연구소 선임연구원
- 2005년 3월 ~ 현재 : 연세대학교 컴퓨터정보통신공학부 부교수
- 주관심분야 : 영상부호화, 영상통신, 디지털 방송, 멀티미디어 통신시스템



김 진 수

- 1991년 : 경북대학교 전자공학과 학사
- 1993년 : KAIST 전기 및 전자공학과 석사
- 1998년 : KAIST 전기 및 전자공학과 박사
- 1997년 ~ 2000년 : 삼성전자 선임연구원
- 2000년 ~ 현재 : 한밭대학교 정보통신컴퓨터공학부 교수
- 주관심분야 : 멀티미디어 스트리밍, SVC, Distributed Video Coding, 디지털 방송

저 자 소 개



김 재 곤

- 1990년 2월 : 경북대학교 전자공학과 학사
- 1992년 2월 : KAIST 전기 및 전자공학과 석사
- 2005년 2월 : KAIST 전기 및 전자공학과 박사
- 1992년 3월 ~ 2007년 2월 : ETRI 선임연구원/팀장
- 2001년 9월 ~ 2002년 11월 : 뉴욕 콜롬비아대학교 연구원
- 2007년 9월 ~ 현재 : 한국항공대학교 항공전자 및 정보통신공학부 조교수
- 주관심분야 : 비디오 신호처리/부호화, 디지털방송 미디어, 미디어 컨버전스, 멀티미디어 통신



정 순 흥

- 2001년 2월 : 부산대학교 전자공학과 학사
- 2003년 2월 : KAIST 전자전산학과 석사
- 2003년 3월 ~ 2005년 3월 : LG전자 주임연구원
- 2005년 4월 ~ 현재 : ETRI 방통미디어연구부 융합미디어연구팀 선임연구원
- 2010년 9월 ~ 현재 : KAIST 전기 및 전자공학과 박사과정
- 주관심분야 : 영상부호화, 멀티미디어방송, 영상통신



배 성 준

- 1997년 2월 : 고려대학교 전자공학과 학사
- 1999년 2월 : KAIST 전기 및 전자공학과 석사
- 2004년 8월 : KAIST 전기 및 전자공학과 박사
- 2004년 8월 ~ 2005년 10월 : 하나로텔레콤 기술전략팀
- 2005년 10월 ~ 현재 : ETRI 스마트TV사업단 선임연구원
- 주관심분야 : IP기반 비디오 부호화 및 전송 기술