정규논문 (Regular Paper)
방송공학회논문지 제18권 제6호, 2013년 11월 (JBE Vol. 18, No. 6, November 2013)
http://dx.doi.org/10.5909/JBE.2013.18.6.872
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

# 오류 강인 SVC 비디오 전송을 위한 Exclusive-OR 기반의 FEC 부호화 시스템 설계 및 성능 분석

이 홍 래[a], 정 태 준[a], 심 상 우[b], 김 진 수[c], 서 광 덕[a]‡

# Design and Performance Analysis of Exclusive-OR Based FEC Coding System for Error Resilient SVC Video Transmission

Hong-rae Lee[a], Tae-jun Jung[a], Sang-woo Shim[b], Jin-soo Kim[c], and Kwang-deok Seo[a]‡

요 약

본 논문에서는 패킷 오류가 발생하는 IP망을 통해 SVC 비디오 전송 서비스를 제공하기 위한 Exclusive-OR 기반의 FEC (forward error correction) 오류제어 시스템을 설계하고 성능을 분석한다. 설계된 시스템에서는 계산적으로 복잡도가 낮은 표준 Exclusive-OR 연산에 기반한 FEC 방법을 활용하고, SVC 비디오의 계층적 구조에 적합하도록 FEC 기법을 적용 한다. 설계된 Exclusive-OR 기반의 오류 제어 시스템의 성능을 검증하기 위하여 NIST-NET 기반의 전송 시뮬레이터를 활용한다. NIST-NET 기반의 시뮬레이터를 통한 SVC 비디오 패킷 전송 실험에 의해 설계된 Exclusive-OR 기반의 FEC 시스템의 오류 강인 전송 성능을 확인한다.

Abstract

In this paper, we design and analyze performance of Exclusive-OR based FEC (Forward error correction) system to deploy SVC video transmission service over packet-loss prone IP network. In the designed system, we adopt standard compliant Exclusive-OR based FEC scheme and apply it to be appropriate to the hierarchical layer structure of SVC video. To verify the performance of the designed Exclusive-OR based FEC system for SVC video transmission, we employ NIST-NET based transport simulator. By the SVC video transmission using the NIST-NET based simulator, we confirm the error resilient transmission performance of the designed Exclusive-OR based FEC system.

Keyword : Forward error correction (FEC), error control, SVC video transmission

a) 연세대학교 (Yonsei University)
b) 안철수연구소 (AhnLab)
c) 한밭대학교 (Hanbat National University)
‡ Corresponding Author : 서광덕(Kwang-deok Seo)
E-mail: kdseo@yonsei.ac.kr
Tel: +82-33-760-2788
※ 이 논문은 2013년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. NRF-2013R1A1A2011635).
이 논문은 2012학년도 연세대학교 학술연구비의 부분적인 지원에 의하여 이루어진 것임.
· Manuscript received September 17, 2013 Revised October 23, 2013 Accepted October 23, 2013

## Ⅰ. 서 론

최근 통신 기술이 발전함에 따라 통신망의 속도도 빠르게 증가해왔다. 유선 통신의 경우 FTTH 기술을 기반으로 각 가정에 초고속 인터넷이 공급되고, 무선 통신의 경우 현재 제공되고 있는 3G 이동통신보다 더욱 빠른 통신 속도를 제공하는 4세대 이동통신(4G)으로 발전함에 따라 소비자들의 고품질 멀티미디어 서비스에 대한 욕구도 증가해 왔다. 유선의 경우 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. 통신망의 속도가 빨라지고 고품질 멀티미디어 서비스에 대한 욕구는 증가해 왔지만, 서비스의 기반이 되는 IP기반의 패킷 네트워크는 안정적인 멀티미디어 서비스를 제공하기 위한 QoS를 제공하지 못한다. 그 이유는 IP망의 경우 패킷 전송이 불안정 하거나 패킷손실, 전송지연, 채널 대역폭 등의 변동이 발생할 가능성이 높고, 패킷 오류를 줄이거나 제어하기 위한 추가적인 기능을 기본적으로 제공하지 못한다[1].

고품질 멀티미디어 서비스의 사용자들은 안정적인 영상 서비스를 제공받기 원한다. 그러나 안정적인 QoS를 제공하지 못하는 IP 기반의 패킷 네트워크에서는 패킷 손실, 전송지연, 채널 대역폭의 변동 등이 발생할 가능성이 높다. 따라서 사용자들의 QoS를 만족시키면서 패킷 오류를 극복할 수 있는 기법이 필요하다[2].

이러한 채널에서의 패킷 손실을 극복하기 위해 일반적으로 링크계층에서 ARQ(Automatic Repeat Request)와 같은 전송 오류 제어를 수행 할 수 있다. 그러나 이러한 접근 방법을 통해 모든 형태의 패킷 오류를 극복할 수는 없으며 상위 계층에서 패킷 손실 및 전송지연의 변동을 유발할 수도 있고, 스트리밍 비디오의 급격한 품질의 열화를 야기 시킬수도 있다. 따라서 채널의 패킷 오류 형태의 변동이 심한 유무선 인터넷 환경에서 성공적으로 비디오를 전송하기 위해서는 응용계층 (application layer)에서 네트워크에 적응적인 전송 오류 제어 기법의 개발이 필요하다. 응용 계층에서의 대표적인 오류제어 방식은 ARQ[3], [4]와 FEC (Forward Error Correction)[5]로 나눌 수 있다. ARQ와 FEC 중 어느것이 더 효율적인지는 사용되어지는 시스템의 특성, 경제성, 장비와 단말의 기능, 사용자의 요구 사항 등에 따라 달라질 수 있다. 그러나 서비스 환경에서 ARQ에 필요한 피드백 채널이 지원되지 않으면 ARQ보다는 FEC가 유리할 것이다. 기본적으로 FEC는 첨가되는 잉여패킷 (parity packet)들로 인해 채널의 대역폭을 낭비시켜 시스템의 효율을 저하시키지만[4] 재전송에 필요한 버퍼와 그에 따른 지연시간을 고려하지 않아도 되기 때문에 멀티캐스트와 같은 실시간 전송에서 발생하는 패킷 오류에 효율적으로 대처할 수 있다. 논문 [6]에서는 패킷 오류가 발생하는 IP 망을 통해 비디오 전송 서비스를 제공하기 위한 효율적인 망 적응적 ARQ 기반의 오류 제어 기법을 제안한다. 그러나, 근본적으로 ARQ 오류제어 기법의 원리를 채택하는 방법이므로 실시간 미디어 전송에는 한계를 나타낼 수 밖에 없다. 특히, SVC와 같은 복잡도가 높은 부호화 기법과 결합되었을 때는 구현 복잡도가 높아 지게 된다.

본 논문에서는 Exclusive-OR 연산을 기반으로한 FEC를 SVC 비디오의 계층적 구조에 적용하기 위한 설계 방안을 제시한다. Exclusive-OR 기반의 FEC는 기존의 Reed-Solomon 또는 BCH (Bose – Chaudhuri – Hocquenghem) 코드 등과 비교했을 때 계산 복잡도가 매우 낮기 때문에 실시간적인 채널 부호화가 필요한 응용에 매우 효율적으로 적용될 수 있다. 특히 SVC와 같은 복잡도가 높은 부호화 기법과 함께 활용되었을 때 그 효과가 배가될 수 있다. 제안된 기법에서는 손실된 패킷에 대한 효율적인 FEC를 수행하기 위해 FEC 패킷을 포함한 수신된 미디어 패킷이 버퍼에 배치되는 방법과 손실된 패킷을 복구하는 메카니즘에 관한 설계를 포함한다.

## Ⅱ. Exclusive-OR 기반의 FEC 알고리듬 개요

실시간 응용에서는 충분한 지연시간을 허용하지 않고, 멀티캐스트와 같은 경우는 패킷 손실이 발생할 경우, 이를 재전송을 통해 복구할 수 있는 피드백 채널을 가지기 힘들

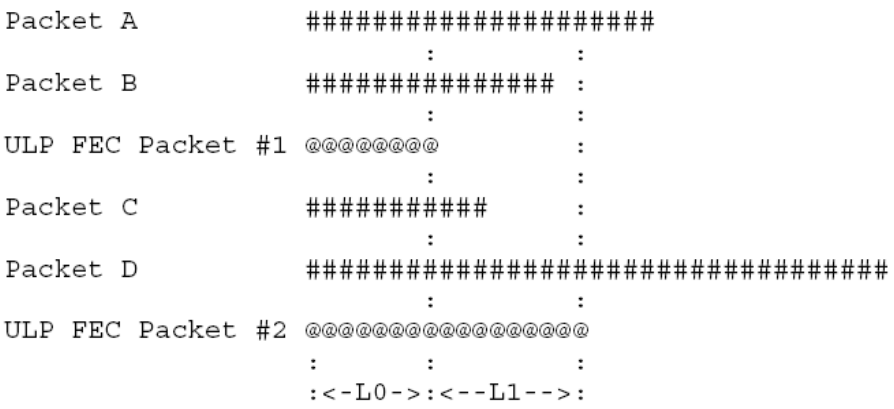



그림 1. Unequal Level Protection 원리
Fig. 1. Principle of Unequal Level Protection

다. 따라서 멀티캐스트 환경에서는 손실된 정보를 복구하기 위한 방법으로 FEC와 같은 기법을 사용한다.

RFC 5109[7]에서는 RFC 2733[8]과 RFC 3009[9]를 확장하여 간단한 Exclusive-OR (XOR) 연산에 기반한 FEC 기법을 표준화하고 있다. RFC 5109에서 정의된 FEC 프로토콜은 보호되는 미디어와 독립적이고 다양한 FEC 설정을 지원할 수 있도록 한다. 또한 서버단에서의 적응적인 설계가 가능하므로 수신단으로부터의 out-of-band 시그널링 신호처리 없이 쉽게 수정될 수 있고, 다양한 종류의 미디어에 대한 FEC 패킷 처리에 응용될 수 있다.

대부분 네트워크 대역폭은 매우 제한적인 반면, 이전 FEC 기법들은 이에 최적화되어 설계되지 않았다. 보통 데이터 스트림의 중요도에 따라 다른 보호 등급을 제공하는 UEP(Unequal Error Protection) 기법을 사용하여 보다 효율적으로 대역폭을 사용하고 있지만, UEP 기법은 보호되는 미디어의 구조에 따라 특정 형식으로 설계되는 경우가 많고, 이를 일반적으로 적용하기에는 무리가 있다.

RFC 5109에서는 UEP 기법을 대체하는 ULP (Unequal Level Protection) 알고리듬을 제시하고 있다[7]. 일반적인 멀티미디어 스트림은 패킷의 뒷부분보다 앞부분에 더 중요한 데이터가 담기는 경향이 있고, 따라서 패킷의 뒷부분보다 앞부분에 더 강한 보호 등급을 적용할 수 있다. 이는 미디어 페이로드 보호의 효율성을 증가시킬 수 있다. ULP 기법은 패킷의 앞부분에 Level에 따르는 더 강한 보호를 적용하고, 뒷부분에 상대적으로 약한 보호를 적용하는 기법이다.

그림 1은 4개의 미디어 패킷으로 2개의 ULP 패킷을 생성하는 예이다. ULP FEC 패킷 #1은 Level 0 (L0)에서 패킷 A와 패킷 B를 보호하고, ULP FEC 패킷 #2는 Level 0 (L0)에서 패킷 C와 패킷 D를 보호하고, Level 1 (L1)에서 패킷 A, B, C, D 를 보호한다. 이는 Level 0에 해당하는 패킷의 앞부분에 더 강한 보호를 제공하는 것이다. 패킷 기반의 FEC를 수행할 경우, 미디어 스트림은 FEC 스트림에 영향을 받지 않는다. 또한 서버에서 FEC 스트림을 미디어 스트림과 분리하여 전송할 경우, 단말 측에서는 FEC 스트림이 마치 없는 것처럼 미디어 스트림을 수신할 수 있다. 이러한 접근은 FEC를 지원하지 않는 단말과 완벽한 호환성을 가지게 할 수 있다. 이때, FEC 기법을 사용하여 생기는 오버헤드는 FEC 패킷이 있을 때만 존재하고, 이는 FEC 패킷의 사용량을 측정하여 쉽게 감시하고 조정할 수 있다.

하나 또는 그 이상의 Level을 가지는 FEC 패킷의 구조는 그림 2와 같다.

첫 12옥텟은 기본적인 RTP[10] 헤더 구간으로써 일반적인 RTP 패킷에 포함되는 헤더필드와 같다. FEC 패킷을 위한 RTP 헤더는 FEC 스트림이 미디어 스트림과 분리되어 전송될 경우에만 사용된다. 기본적인 설정 값은 RFC 3550[10]을

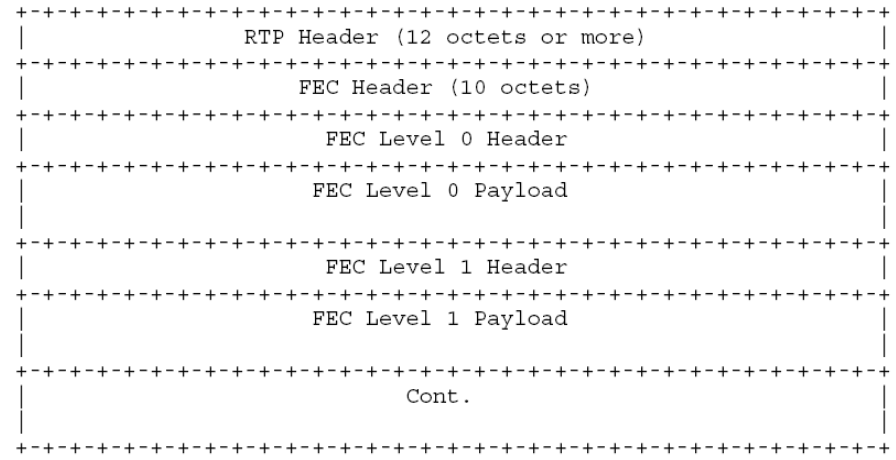



그림 2.. FEC 패킷 구조
Fig. 2. FEC packet structure

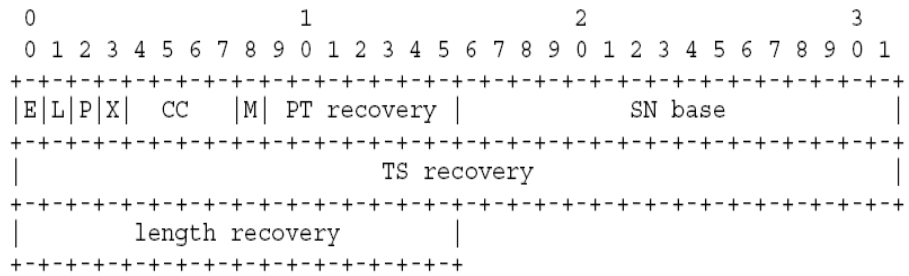



그림 3. FEC 헤더 포맷
Fig. 3. FEC header format

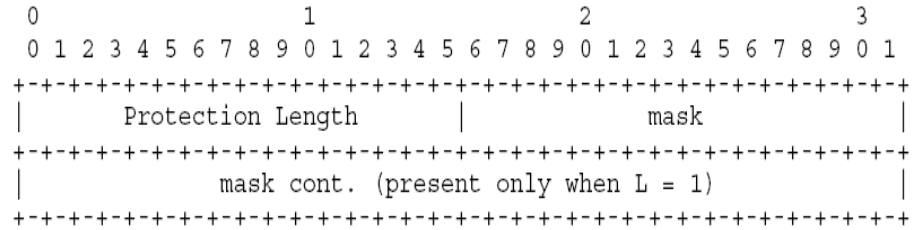



그림 4. ULP Level 헤더 포맷
Fig. 4. ULP Level header format

따른다. 다음 10옥텟은 FEC 헤더 구간으로 헤더 구성 필드는 그림 3과 같다.

그림 4는 FEC Level 헤더를 구성하는 필드이다. 원하는 Protection length에 따라서 mask의 크기가 결정된다. FEC Level 헤더는 Long mask의 사용 여부에 따라 4 옥텟 또는 8 옥텟으로 구성된다.

## III. 멀티캐스트 SVC 서버/단말 구조 및 FEC 동작 설계

### 1. 멀티캐스트 SVC 비디오 서버 및 단말 구조

그림 5는 SVC [11] 비디오 전송을 위한 멀티캐스트 서버

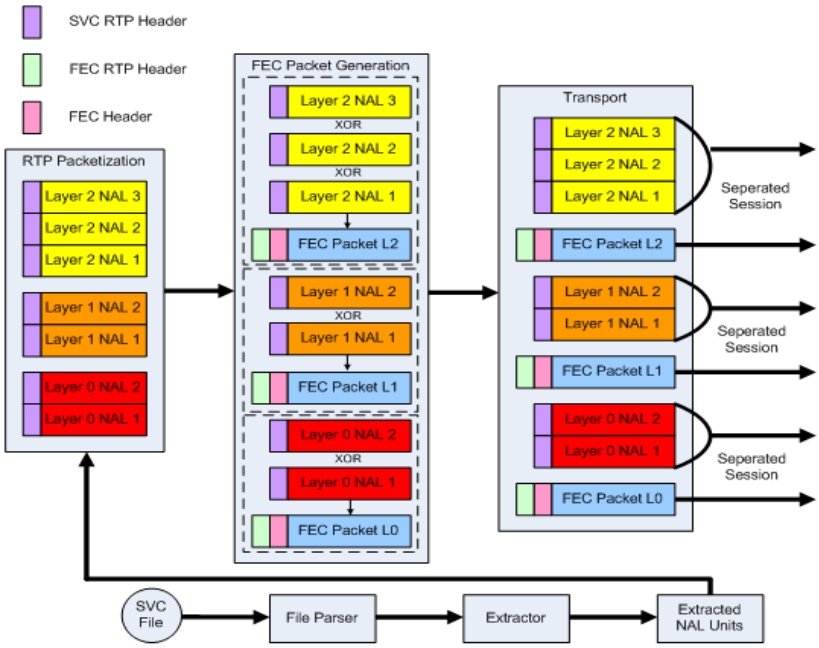



그림 5. 멀티캐스트 서버 구조
Fig. 5. Multicast server structure

의 기본적 동작 구조이다. 이 예에서는 중요도가 높은 Layer0와 Layer1의 경우 2개의 NAL unit을 묶어서 XOR 연산을 적용하며 상대적으로 중요도가 낮은 Layer2에 대해서는 3개의 NAL unit을 묶어서 XOR 연산을 적용한다. 각

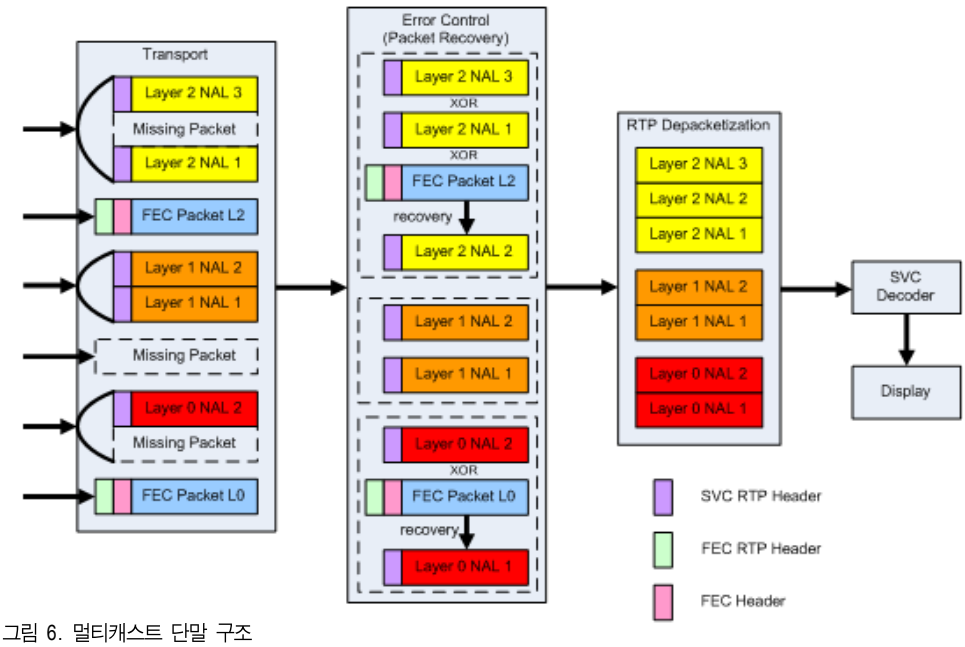



그림 6. 멀티캐스트 단말 구조
Fig. 6. Multicast terminal structure

Layer 마다 미디어 스트림 전송을 위한 세션과 FEC 패킷을 전송하는 세션이 독립적인 RTP 세션을 통해 생성된다.

그림 6은 그림 5의 서버에 대응하는 멀티캐스트 단말의 구조를 나타낸다. 손실된 패킷이 존재하는 경우 안전하게 수신된 패킷을 활용하여 XOR 연산을 통해 손실된 패킷을 복원해내게 된다.

## 2. FEC 인코딩 동작 설계

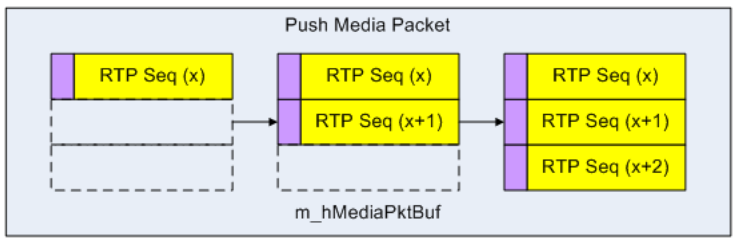
FEC 인코딩 과정은 기본적으로 멀티캐스트 서버에서 수행되며, 미디어 패킷과 생성된 FEC 패킷은 서로 분리된 채널로 전송된다. 그림 7은 FEC 인코딩이 이루어지는 과정을 보여준다. 그림 7에서 보여주는 예는 미디어 패킷 3개당 하나의 FEC 패킷이 생성되고, 미디어 패킷의 Base Sequence Number(Base SN)가 x라고 가정한다.

FEC를 위해 필요한 미디어 패킷 개수에 따라 미디어 패킷을 위한 버퍼를 설정한다. 버퍼에 필요한 미디어 패킷 개수만큼 Push되면 첫번째 미디어 패킷의 SN을 Base SN으로 하여 FEC 패킷을 생성하고, 생성된 FEC 패킷을 FEC 인코딩의 출력으로 내놓는다. 이때 미디어 패킷으로 FEC 패킷을 만드는 방식은 XOR 연산을 기반으로 한 RFC 5109를 따른다[9].

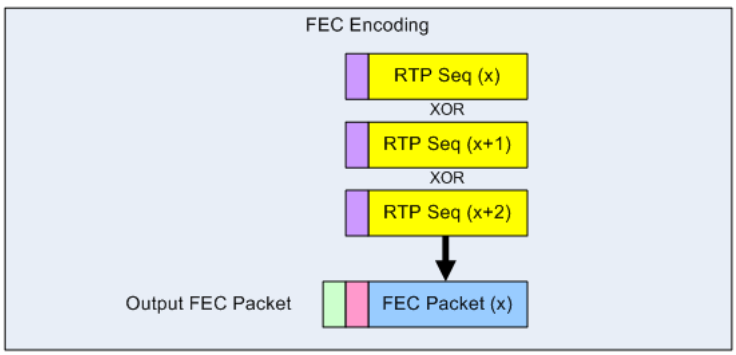
## 3. FEC 디코딩 동작 설계

멀티캐스트 서버는 미디어 패킷과 FEC 패킷을 분리된 채널로 전송하고, 따라서 수신 측에서 미디어 패킷과 FEC 패킷은 기본적으로 동기화를 맞추기 어렵다. 또한 미디어 패킷의 손실이 있을 경우, 해당 FEC 패킷을 찾아 손실된 미디어 패킷을 복구하는 과정을 거쳐야 한다.

그림 8은 미디어 패킷이 먼저 도착한 상태에서 FEC 패킷을 Push한 경우, 손실된 미디어 패킷을 복구하는 과정이다. 수신된 미디어 패킷과 FEC 패킷을 저장하기 위한 버퍼를 사용한다. 이들 버퍼는 Linked List로 구성된 Queue로서,

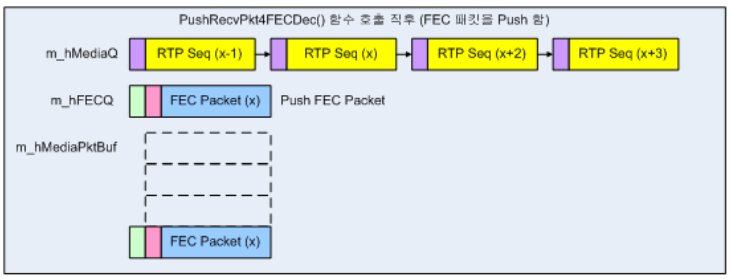



(a) 미디어 패킷을 Push하는 과정

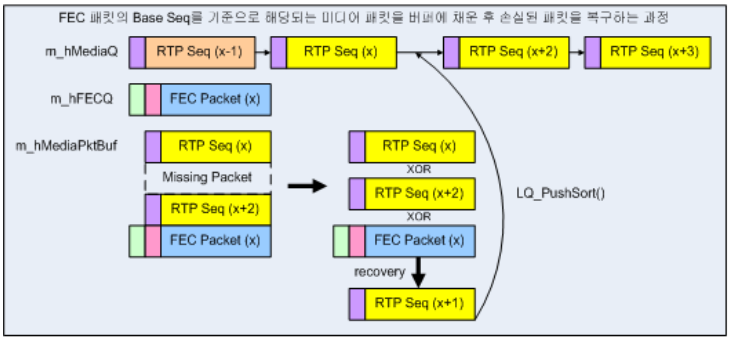



(b) 3개의 미디어 패킷으로 FEC 패킷을 생성하는 과정

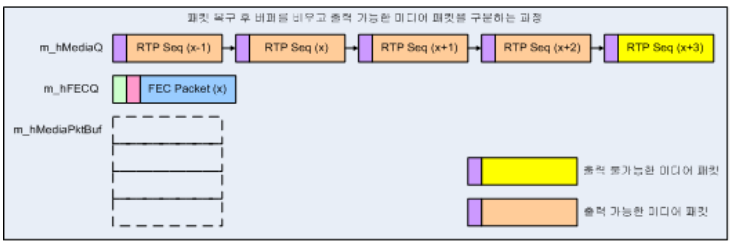
그림 7. FEC 인코딩이 이루어지는 과정
Fig. 7. Process of FEC encoding



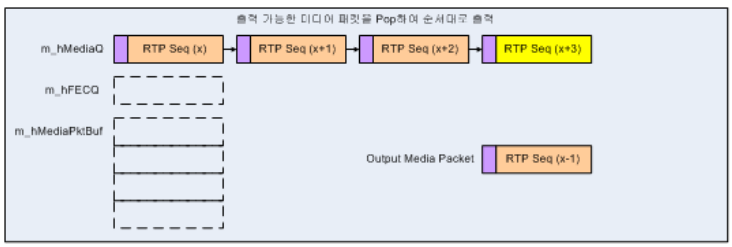

(a) FEC 패킷을 Push하면서 FEC 디코딩을 호출




(b) FEC 패킷의 Base SN을 기준으로 해당되는 미디어 패킷을 버퍼에 채운 후 손실된 패킷을 복구하는 과정



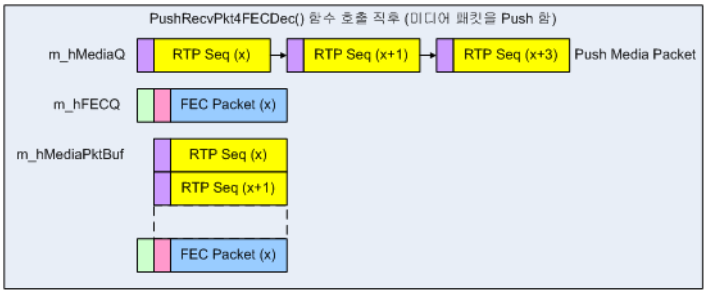

(c) 패킷 복구 후 버퍼를 비우고 출력 가능한 미디어 패킷을 구분하는 과정



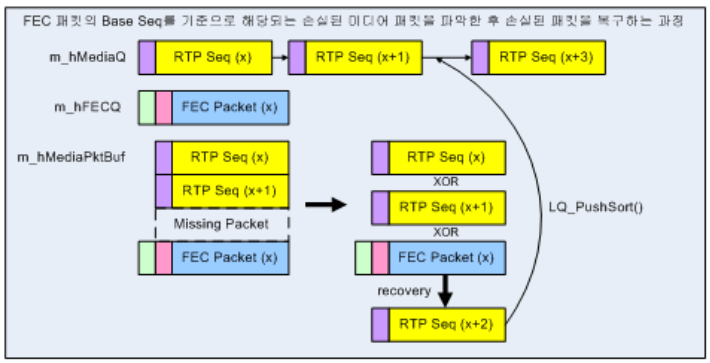

(d) 출력 가능한 미디어 패킷을 Pop하여 순서대로 출력

그림 8. 미디어 패킷이 먼저 도착한 상태에서 FEC 패킷을 Push한 경우 FEC 디코딩이 이뤄지는 과정
Fig. 8. FEC decoding process for pushing FEC packet while media packets arrived earlier

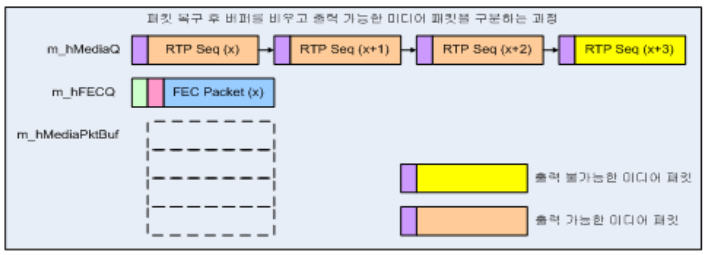
패킷들은 RTP 헤더의 SN을 기준으로 정렬되어 삽입된다. FEC 패킷의 Base SN은 x이고 버퍼에는 SN이 x+1에 해당하는 미디어 패킷이 손실된 상태로 SN이 x-1부터 x+3인 미디어 패킷이 저장되어 있다고 가정한다. 그림 8 (a)는 FEC 패킷을 Push하면서 FEC 디코딩이 이뤄질 때 버퍼의 상황을 보여준다. (a)에서 SN이 x+1에 해당하는 미디어 패



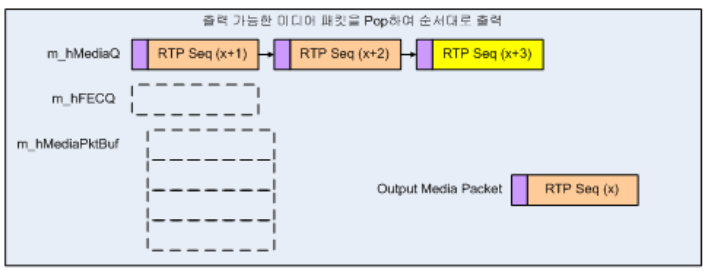

(a) 미디어 패킷을 Push하면서 FEC 디코더를 호출한 직후




(b) FEC 패킷의 Base SN을 기준으로 해당되는 미디어 패킷을 버퍼에 채운 후 손실된 패킷을 복구하는 과정




(c) 패킷 복구 후 버퍼를 비우고 출력 가능한 미디어 패킷을 구분하는 과정




(d) 출력 가능한 미디어 패킷을 Pop하여 순서대로 출력

그림 9. FEC 패킷이 먼저 도착한 상태에서 미디어 패킷을 Push한 경우 FEC 디코딩이 이루어 지는 과정
Fig. 9. FEC decoding process for pushing media packet while FEC packets arrived earlier

킷이 손실된 것을 확인할 수 있고, 현재 Push된 FEC 패킷의 Base SN이 x이므로 이 FEC 패킷을 이용하여 손실된 미디어 패킷을 복구할 수 있다. (b)에서는 FEC 패킷의 Base SN을 기준으로 해당 미디어 패킷을 버퍼에 채운 후, 손실된 미디어 패킷을 복구하고, 복구된 패킷을 미디어 버퍼에 정렬하여 삽입하는 과정을 보여준다. 이때, SN이 x-1에 해

당하는 미디어 패킷은 현재 Push된 FEC 패킷의 Base SN보다 작으므로 손실 여부에 상관없이 출력 가능한 상태가 된다. (c)는 패킷 복구 후 미디어 버퍼를 비우고 출력 가능한 미디어 패킷을 구분하는 과정이다. FEC 패킷의 Base SN에 해당하는 x부터 FEC를 위해 요구되는 패킷 개수에 따라 x+2까지는 출력 가능한 상태가 된다. (d)에서는 출력 가능한 미디어 패킷을 Pop하여 출력으로 내놓는 과정을 보여준다. FEC 디코딩을 하면 한번 호출할 때마다 하나의 출력 가능한 미디어 패킷을 얻을 수 있고, 출력 가능한 미디어 패킷이 없을 경우에는 출력이 없다.

그림 9는 FEC 패킷이 먼저 도착한 상태에서 미디어 패킷을 Push한 경우 FEC 디코딩 시 손실된 미디어 패킷을 복구하는 과정을 보여준다. FEC 패킷의 Base SN은 x이고, 버퍼에는 SN이 x 에 해당하는 미디어 패킷과 x+1에 해당하는 미디어 패킷이 저장되어 있다. 현재 Push되는 미디어 패킷의 SN은 x+3이고, 따라서 SN이 x+2에 해당하는 미디어 패킷이 손실되었다고 가정한다.

그림 9 (a)는 FEC 패킷을 Push하면서 FEC 디코딩이 이뤄질 때 버퍼의 상황을 보여준다. 먼저 도착한 FEC 패킷의 Base SN에 따라 버퍼에는 해당 미디어 패킷이 채워져 있고, SN이 x+2에 해당하는 미디어 패킷을 기다리는 상황에서 SN이 x+3에 해당하는 미디어 패킷이 Push되었다. 따라서 SN이 x+2에 해당하는 미디어 패킷이 손실되었다고 판단하고, 이를 복구하게 된다. (b)에서는 손실된 미디어 패킷을 복구하고, 복구된 패킷을 정렬하여 삽입하는 과정을 보여준다. (c)는 패킷 복구 후 미디어 버퍼를 비우고 출력 가능한 미디어 패킷을 구분하는 과정이다. FEC 패킷의 Base SN에 해당하는 x부터 FEC를 위해 요구되는 패킷 수에 따라 x+2까지는 출력 가능한 상태가 된다. (d)에서는 SN이 x에 해당하는 출력 가능한 미디어 패킷을 Pop하여 출력으로 내놓는 과정을 보여준다.

## IV. 성능평가

실제적인 네트워크 환경에서 제안된 방법의 성능을 평가하기 위하여 IP 네트워크 에뮬레이터인 NIST-Net을 활용하여 실험을 하였다[12]. NIST-Net을 이용하면 일정한 대역폭을 가지는 가상 네트워크를 생성할 수 있고, 가상 네트워크에서 사용되는 라우터는 DRD(derivative random drop) 메커니즘을 기반으로 채널에서의 패킷 손실을 발생시키고 패킷 손실률을 조절할 수 있다[13]. 본 실험에서는 NIST-Net

표 1. Exclusive-OR 기반의 FEC에 의한 패킷 복구율
Table 1. Packet reconstruction ratio by Exclusive-OR based FEC

(단위 : %)

| FEC를 위해 필요한 미디어 패킷 개수 | | | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| 생성된 FEC 패킷 개수 | | | 5000 (50%) | 3333 (33%) | 2500 (25%) | 2000 (20%) |
| 복구되지 않은 패킷 개수 | 네트워크 패킷 손실률 | 1% | 1.8 | 2.9 | 3.8 | 4.9 |
| | | 2% | 7.4 | 11.5 | 15.9 | 18.7 |
| | | 3% | 17.8 | 26.7 | 32.6 | 43.3 |
| 복구되지 않은 패킷 비율(%) | | 1% | 0.02 | 0.03 | 0.04 | 0.05 |
| | | 2% | 0.07 | 0.12 | 0.16 | 0.19 |
| | | 3% | 0.18 | 0.27 | 0.33 | 0.43 |
| 손실된 패킷 가운데 복구된 패킷 비율 (%) | | 1% | 98.16 | 97.13 | 96.24 | 95.12 |
| | | 2% | 96.32 | 94.26 | 92.01 | 90.79 |
| | | 3% | 94.08 | 91.2 | 89.13 | 85.74 |

에 의해 생성된 가상 IP 네트워크를 통하여 SVC 비디오를 전송하면서 패킷 오류 제어에 관련된 시뮬레이션을 수행하였다. 그런데, NIST-Net 에뮬레이터는 멀티캐스트 실험 환경을 제공하지 않는다. 따라서 Exclusive-OR 기반 FEC 기법의 경우, 모듈에 자체적인 패킷 손실 API (application programming interface)를 추가하여 송신부에서 지정된 손실 확률에 따라 패킷을 손실시켜 전송하고, 수신부는 수신된 패킷들을 이용하여 손실된 패킷을 복원하는 방법으로 실험을 진행하였다.

성능 평가를 위해 SVC NAL 유닛을 포함하는 10000개의 미디어 패킷으로 100번씩 실험하여 패킷 복구율의 평균값을 구하였다.

표 1은 FEC를 위해 필요한 미디어 패킷 개수에 따라 1~3%의 패킷 손실율에 대한 복구되지 않은 패킷 개수, 복구되지 않은 패킷 비율 및 손실된 패킷 가운데 최종적으로 복구된 패킷 비율을 표로 나타낸 결과이다.

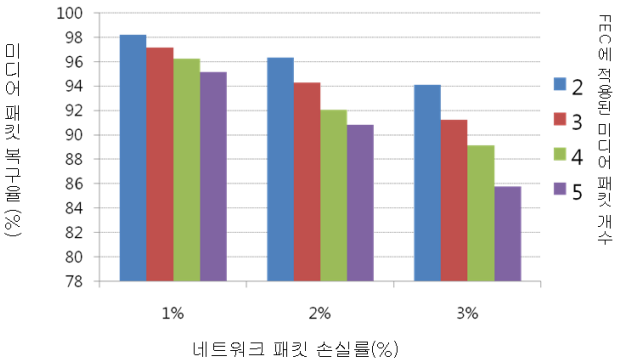



그림 10. FEC에 적용된 미디어 패킷 개수에 따른 최종적인 미디어 패킷 복구율
Fig. 10. Final media packet reconstruction ratio as to variation of number of media packets applied in FEC

그림 10은 표 1의 실험 결과에서 손실된 패킷 가운데 최종적으로 복구된 미디어 패킷의 비율을 도표로 나타낸 결과이다. FEC에 적용된 미디어 패킷의 개수가 작으면 작을수록 오류에 대해 보다 더 강인함을 알 수 있다. Exclusive- OR 연산에 기반한 FEC이므로 FEC에 적용된 미디어 패킷의 개수 중에서 2개 이상이 손실될 경우 복구하기 어렵다. 즉 패킷의 연속손실(Burst)이 발생한 경우가 이에 해당한다.

또한 패킷 손실률이 작을수록 패킷 복구율이 더 높게 나온다. 패킷 손실률이 2%까지는 90% 이상의 복구율을 보인다. 그러나 패킷 손실률이 3%로 늘어나면 복구율은 급격히 떨어진다.

FEC 디코딩의 성공률이 90%를 넘는다고 해서 모든 패킷이 복원되는 것은 아니다. 미디어 패킷의 복원에 실패하여 디코더에 해당 NAL 유닛 데이터를 전달하지 못하면 디코더는 디코딩을 중단하고 오류가 발생한다. 따라서 약간의 NAL 유닛 데이터가 손실되더라도 디코딩을 할 수 있는 디코더의 구현 또한 필요하다.

본 논문에서 설계된 시스템에서는 미디어 패킷이 손실되면 FEC 패킷을 이용하여 복구하지만, 미디어 패킷이 손실되지 않을 경우 FEC 패킷은 해당 채널에서 잉여 패킷 (Parity Packet)으로 간주된다. 이는 통신망에 오버헤드 (Overhead)가 되어 통신망의 대역폭을 추가로 사용한다는 단점을 나타낸다. 그러나 재전송 요청을 위한 피드백 채널이 필요하지 않고, 재전송 요청 및 처리를 위한 대기시간을 요구하지 않으므로 실시간 전송 특성을 갖는 멀티캐스트와 같은 전송 시스템에 적합하다. 또한 유니캐스트 전송 환경에서 별도의 피드백 채널을 가지지 못하는 경우에도 유용하게 사용된다.

FEC를 적용할 경우 FEC 패킷의 생성 및 손실 패킷의 복구 시간에 의해 추가적인 지연시간이 발생한다. 그러나 Exclusive-OR 기반의 FEC는 매우 간단한 연산을 바탕으로 작동하므로 지연시간이 무시할 정도로 매우 작고, 연산 과정이 단순하여 많은 응용 분야에 쉽게 사용될 수 있다는 장점이 있다. 기존 Reed-Solomon (RS) FEC 알고리듬의 경우와 비교하면 Interleaving 및 FEC 정보를 생성하는 높은 계산 복잡도를 요구하는 RS방식 FEC 대신에 단순한 계산 복잡도를 갖는 XOR 연산만을 이용해서 IP망에서 현실적인 패킷 손실 확률 범위인 3% 이내의 패킷 손실에 대해서 매우 실용적이고 효과적으로 적용할 수 있다.

## V. 결 론

본 논문에서는 Exclusive-OR 기반의 FEC를 통하여 IP 기반의 패킷 네트워크에서 패킷 손실, 패킷 지연, 망의 대역폭 변화 등이 발생했을 경우에 대응할 수 있는 Exclusive-

OR FEC 기반의 오류 강인 SVC 비디오 전송 시스템 구현을 위한 설계 방안을 제시하였다. 특히, 구현 상 어려운 부분인 미디어 패킷과 FEC 패킷 중 어떠한 패킷이 손실되었는가에 따라 다양한 복구 시나리오에 기반한 시스템 설계 방법을 제시하였다. 또한 패킷의 앞부분에 영상의 중요한 정보를 많이 담고 있는 특성을 이용하여 UEP (Unequal Error Protection)대신 ULP (Unequal Level Protection)를 사용하였다.

실험을 통해 인터넷 망에서의 현실적인 패킷 손실률 범위인 3% 이내의 낮은 패킷 손실률을 나타내는 IP망을 통한 전송 환경에서 작은 개수의 미디어 패킷을 FEC 부호화 단위로 설정하게 될 경우에도 높은 복구율을 얻을 수 있음을 검증할 수 있었다.

본 논문에서 제안한 SVC 비디오 전송을 위한 Exclusive-OR 기반의 FEC 부호화 및 전송 구조는 다양한 응용의 스트리밍 서비스에 적용될 수 있다. IP 망을 통한 VOD와 같은 유니캐스트 서비스, IPTV와 같은 멀티캐스트 또는 브로드캐스트 서비스에도 활용이 가능하다.

## 참 고 문 헌(References)


[1] 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.

[2] 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.

[3] J. Ott, S. Wenger, N. Sato, C. Burmeister, J. Rey, "Extend RTP profile for real-time transport control protocol (RTCP)-based feedback (RTP/AVPF)," RFC4585, July 2006.

[4] J. Rey, D. Leon, A. Miyazaki, V. Varsa, R. Hakenberg, "RTP retransmission payload format," RFC4588, July 2006.

[5] M. Watson, "Basic Forward Error Correction (FEC) Schemes," RFC5445, March 2009.

[6] S. Shim, et. al., "Network adaptive ARQ error control scheme for effective video transport over IP networks," Journal of Broadcast Engineering, vol. 16, no. 3, May 2011.

[7] A. Li, "RTP payload format for generic forward error correction," RFC5109, December 2007.

[8] J. Rosenberg, H. Schulzrinne, "An RTP payload format for generic forward error correction," RFC2733, December 1999.

[9] J. Rosenberg, H. Schulzrinne, "Registration of parity FEC MIME types," RFC3009, November 2000.

[10] H. Schulzrinne, S. Casner, R. Federick, V. Jacobson, "RTP: A transport protocol for real-time applications," RFC3550, July 2003.

[11] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," IEEE Trans. Circuits and Syst. for Video Technol., vol.17, no.9, pp.1103-1120, Sep. 2007.

[12] NIST-Net, Software provided by National Institute of Standards and Technology (NIST), http://www-x.antd.nist.gov/nistnet.

[13] M. Gaynor, "Proactive packet dropping methods for TCP gateways," Available at http://www.eecs. harvard.edu/~gaynor/final.ps, Oct. 1996.


## 저 자 소 개


이 홍 래

- 2010년 2월 : 연세대학교 컴퓨터정보통신공학부 학사
- 2012년 8월 : 연세대학교 전산학과 석사
- 2012년 9월 ~ 현재 : 연세대학교 전산학과 박사과정
- 주관심분야 : 영상부호화, 영상통신, 멀티미디어 통신 프로토콜

## 저 자 소 개

### 정 태 준

- 2010년 2월 : 연세대학교 컴퓨터정보통신공학부 학사
- 2012년 8월 : 연세대학교 전산학과 석사
- 2012년 9월 ~ 현재 : 연세대학교 전산학과 박사과정
- 주관심분야 : 영상부호화, 영상통신, 멀티미디어 통신 프로토콜

### 심 상 우

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

### 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 정교수
- 주관심분야 : 멀티미디어 스트리밍, SVC, Distributed Video Coding, 디지털 방송

### 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
- 2012년 9월~2013년 8월 : Courtesy Professor, Univ. of Florida, USA
- 2005년 3월 ~ 현재 : 연세대학교 컴퓨터정보통신공학부 정교수
- 주관심분야 : 영상부호화, 영상통신, 디지털 방송, 멀티미디어 통신시스템