

멀티미디어 환경에서 MPEG 데이터의 효율적 재전송 기법

(An Efficient Retransmission Method for MPEG Streams in Multimedia Applications)

이석훈 [†] 이승룡 [‡]

(Seok-Hoon Lee) (Sungyoung Lee)

요약 연속성과 대용량의 특성을 가진 멀티미디어 데이터를 효율적으로 전송하기 위해서 MPEG과 같은 압축기법을 많이 사용한다. 하지만, MPEG은 재생시 서로 다른 종류의 프레임들을 상호 참조하기 때문에 어떤 특정 프레임에서 발생한 오류가 다른 프레임으로 전파되는 특성을 가지고 있다. 한편, 멀티미디어 응용을 위한 통신 시스템에서 전송 시 오류가 발생한 데이터나 지연이 큰 데이터는 시간적 제약 때문에 일반적으로 재전송을 하지 않고 폐기해 버린다. 그러나 임무가 치명적인 멀티미디어 응용과 같이 재전송이 필요한 환경에서 전송 시 오류가 발생한 MPEG 데이터 프레임을 단순히 폐기해 버리는 경우 이를 참조하는 다른 프레임들의 연속적인 손실을 유발하여 QoS의 급격한 저하를 야기시킬 수 있다. 따라서, 본 논문에서는 재전송이 필요한 멀티미디어 응용에서 재전송 회수를 줄이면서도 QoS와 데이터의 연속성을 지원할 수 있는 선택적 재전송 기법을 제안한다. 제안한 기법은 전송전에 프레임의 크기와 종류에 대한 정보를 미리 가지고 있는 가상버퍼를 사용하여 전송중 프레임 손실로 인한 오류 전파 범위를 파악하고, 복구가 가능한 데이터만을 선택적으로 재전송 함으로써 재전송 회수를 최소화하는 방법이다. 시뮬레이션 연구결과 제안한 기법은 오류가 발생한 모든 데이터를 전부 재전송하는 경우에 비하여 거의 유사한 품질의 QoS를 유지하면서도 재전송에 따른 오버헤드를 감소시켰다.

Abstract Since multimedia data are characterized by continuity and massive volume, compression techniques such as MPEG are used for efficient transmission. However, a transmission error occurring at a frame tends to be propagated to other frames because MPEG frames cross-reference one another upon presentation. Due to time constraints, it is usual to discard without retransmission the data which have transmission errors or which cause serious delay. But in the cases of multimedia applications where retransmission is required (for example, mission critical applications), the QoS may be severely deteriorated by the continuous loss caused by propagation if damaged frames are simply discarded. In this paper, we propose a selective retransmission method where the number of retransmissions can be reduced without losing QoS and data continuity. Since the proposed method keeps track of error propagation ranges using virtual buffers which have information about the sizes and types of frames before transmission, the number of retransmissions from the transmitter to the receiver can be minimized by selecting and retransmitting the recoverable data only. Simulation study show the proposed method reduces retransmission cost considerably while maintaining QoS similar to the case of unconditionally retransmitting all the damaged data.

1. 서론

멀티미디어 응용을 위한 고속통신 시스템에서 오류가

발생한 데이터나 지연이 큰 전송 데이터는 시간적 제약 때문에 일반적으로 재전송을 하지 않고 폐기해 버린다. 그러나 재전송을 통해 오류를 복구해 줄 수 있는 환경에서는 재전송을 수행하여 높은 서비스 품질 (QoS: Quality of Service)을 유지하는 것이 바람직하며[3], 또한 임무가 치명적인 멀티미디어 응용 [6]과 같이 사

[†] 비회원 경희대학교 전자계산공학과

[‡] 종신회원 경희대학교 전자계산공학과 교수

논문접수: 1997년 1월 28일

심사완료: 1997년 10월 27일

용자가 원하는 QoS를 보장받기 위해서는 재전송 기법이 필요하다. 예를 들어 환자의 암세포 위치나 전이 유무를 원격으로 검진하는 경우 공교롭게도 암세포 부분의 데이터가 전송 시 손실을 입어 의사가 오진을 하게 되면 치명적인 문제를 유발시킬 수 있다.

한편, MPEG (Motion Picture Expert Group)은 공간적인 중복성 뿐만 아니라 시간적인 중복성까지 제거하여 높은 압축율을 얻을 수 있어 동영상 데이터의 표준 압축방법으로 많이 사용되고 있다 [2]. MPEG은 독립적으로 재생이 가능한 I (Intra-mode) 프레임과, 앞의 I 또는 P (Predicted-mode) 프레임을 참조하는 P 프레임, 앞뒤의 I 또는 P 프레임을 양방향 참조하는 B (Bidirectional mode) 프레임의 3가지 종류의 프레임들이 일정한 순서에 따라 배치되는 특성을 가지고 있다. 때문에 참조가 시작되는 앞의 I 프레임 또는 P 프레임에 오류가 발생하게 된다면 이의 영향이 뒤의 여러 개의 다른 프레임들까지 전파(propagation)되어 연속적인 손실을 유발할 수 있다. 프레임 손실 문제를 해결하기 위한 방법으로 셀 손실 은닉(cell loss concealment) 또는 전 방향 에러 복구(forward error correction)등의 방법들이[5] 사용되고 있다. 그러나 이러한 방법들은 손실된 데이터로 인한 화질의 급격한 품질저하 부분을 사용자가 인지하지 못하도록 은닉해 주거나 복구해주는 기법으로 재전송으로 인해 발생할 수 있는 지연은 회피 할 수 있으나 대신 완벽하게 복구 해 주지 못한다. 때문에 위에서 예로 들은 임무가 치명적인 원격 의료진료 응용과 같은 경우에는 유효한 해결방법이 될 수 없다.

따라서 본 논문에서는 재전송이 필요한 멀티미디어 응용에서 재전송 비용을 줄이고 QoS와 데이터의 연속성을 보장하기 위한 선택적 재전송 기법을 제안한다. 이를 위하여, 본 논문에서는 프레임의 크기 및 종류 등과 같이 재생될 데이터에 대한 정보를 미리 유지해서 앞으로 수신할 데이터에 대한 성질을 예측할 수 있는 메카니즘인 가상버퍼를 도입하였다. 가상버퍼를 이용하면 전송 시 어떤 프레임에 오류가 발생했는지를 알 수 있으며 오류의 영향이 자신에만 국한되는지 아니면 다른 프레임들까지 전파되는지의 여부를 판단할 수 있어 오류 복구가 가능한 경우에만 재전송을 하는 선택적 재전송이 가능하게된다. 이같은 선택적 재전송 기법은 오류가 발생한 모든 프레임을 재전송하는 경우와 유사한 서비스 품질을 획득할 수 있을 뿐만 아니라 적은 회수의 재전송으로 시스템의 오버헤드를 감소시킬 수 있다. 가상버퍼를 위한 프레임 정보는 전송 측에서 MPEG 데이터의 인코딩 또는 사전처리 시 독립된 파일의 형태로

미리 구성해 둘 수 있다. 재전송 요구는 멀티미디어 응용 시 재생, 정지, 후진 등의 제어 명령들을 보내기 위해 사용하는 기존의 제어채널을 이용할 수 있으며, 가상 버퍼를 위한 일련의 프레임 정보와 재전송 되는 데이터는 협대역의 재전송 채널을 새로 설립하여 사용한다. 제안한 선택적 재전송 기법은 네트워크상의 지연이 작거나 수신 측에서 충분한 크기의 버퍼로서 재전송까지의 지연을 흡수할 수 있는 경우에 활용이 가능하다.

본 논문의 구성은 다음과 같다. 2 장에서는 MPEG의 특징을 간단히 소개하며, 3 장에서는 제안한 시스템 모델과 본 논문에서의 가정을 기술한다. 4 장에서는 가상 버퍼를 이용한 오류 검출 및 선택적 재전송 기법을 기술하고, 5 장에서는 제안한 알고리즘의 시뮬레이션 결과를 보여주며, 마지막으로 6 장에서 결론을 맺는다.

2. MPEG의 특징

MPEG-2에서는 공간축의 중복성을 제거하기 위해서 DCT (Discrete Cosine Transform), 시간축의 중복성을 제거하기 위해서 움직임 보상방법 (motion compensation)을 사용하며 이를 위해서 I, P, B의 세 가지 형태의 프레임을 정의한다. I 프레임은 DCT만을 이용하여 인트라 코딩(intra-coding)한 프레임으로서 다른 프레임을 참조하지 않고 독립적으로 디코딩이 가능하다. P 프레임은 움직임 보상을 이용하여 자신의 앞에 위치한 I 프레임 또는 P 프레임을 참조하여 디코딩 된다. B 프레임은 움직임 보상을 이용하여 디코딩 되지만 자신의 앞, 뒤에 위치한 I 프레임과 P 프레임, 또는 P 프레임을 양방향으로 참조하여 재생되는 특징을 가지고 있다 [그림 1].

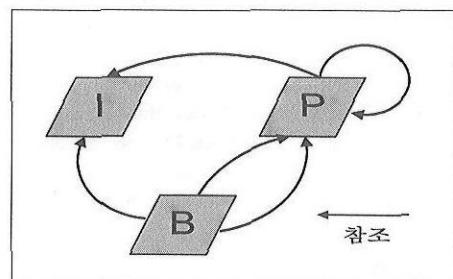


그림 1 I, P, B 프레임의 상관 관계

따라서 MPEG에서는 프레임들이 [그림 2-a]의 형태로 인코딩되며, I 프레임으로부터 다음번 I 프레임 직전의 B 프레임까지를 GOP (Group Of Picture)라 정의를

하고 I 프레임과 P 프레임, P 프레임과 P 프레임이 나타나는 주기를 M이라 정의한다.

MPEG 테이터는 GOP의 연속적인 주기로서 구성되며 급격한 장면전환 (scene change)이 발생하여 움직임 보상으로 완전한 재생이 어려운 경우에는 P 또는 B 프레임이 와야 할 곳에 강제로 I 프레임을 삽입하여 인트라 코딩을 한다. 그러나 P 프레임이나 B 프레임과 같은 움직임 보상방법을 통해 재생되는 프레임들은 자신이 재생되기 이전에 참조할 프레임들이 미리 있어야 하기 때문에 실제 재생을 위해 전송되거나 저장될 경우에 순서는 [그림 2-b]와 같이 바꿔야 한다. 즉 [그림 2-a]에서 2번의 B 프레임이 재생되기 위해서는 이 프레임이 참조하는 앞쪽의 I 프레임 뿐만 아니라 뒤쪽의 P 프레임까지 미리 도착해 있어야 하기 때문에 뒤의 P 프레임이 B 프레임보다 재생순서가 늦어도 B 프레임의 앞쪽에 미리 도달해 있어야만 한다.

이러한 MPEG의 특성에 의거하여 각각의 프레임들의 중요도에 따른 우선 순위를 부여할 수 있다. [그림 3]은 GOP가 12이고 M이 3인 MPEG 데이터의 예이다. 이 경우에 만일 I_1 프레임에 오류가 발생하게 되면 재생 시 이 오류는 I_1 뿐만이 아니라 B_3 부터 B_{12} 프레임들에 전파되어 모두 14개의 프레임에 영향을 미치게 된다. 또한 P_2 프레임에 오류가 발생하면 이를 참조하는 뒤의 프레임들, 즉 B_5 에서 B_{12} 까지 영향을 미치게 된다. 이러한 특징을 고려하여 오류 발생 시 GOP안에서 프레임의 중요도에 따른 제거 우선순위를 [그림 3]과 같이 결정할 수 있다.

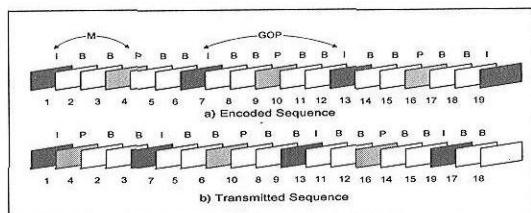


그림 2 인코딩시와 전송시의 프레임의 순서 ($M=3$, $GOP=6$)

이때 낮은 우선 순위의 프레임이 오류발생 시 가장 먼저 제거된다. B 프레임은 다른 프레임에 의해 영향을 받기만 하고 영향을 미치지는 않으며 또한 일반적인 경우 비디오에서 한 프레임 정도의 손실은 사용자가 인지 할 수 없으므로 가장 먼저 제거된다. 그러나 연속적인 손실은 바람직하지 않기 때문에 인접한 B 프레임의 우선순위를 다르게 결정하였다. 그리고 P 프레임은 재생

시 앞쪽에 있는 I 또는 P 프레임을 참조하기 때문에 GOP의 뒤로 갈수록 우선순위가 낮다.

Frame Type	P_1	B_1	B_2	I_1	B_3	B_4	P_2	B_5	B_6	P_3	B_7	B_8	P_4	B_9	B_{10}	I_2	B_{11}	B_{12}	P_5
Frame Priority	6	5	1	6	5	2	6	5	3	6	5	4	6	5	1	6	5	2	

그림 3 GOP에서 Frame의 우선순위

MPEG의 I, B, P 프레임들은 그 압축방식의 차이에 의해서 각각의 프레임 크기가 다르다. 즉 I 프레임은 움직임 보상을 하지 않고 DCT만으로 인코딩 되기 때문에 그 크기가 P, B프레임에 비해서 상대적으로 크며, P, B 프레임들은 움직임 보상 방법을 통해서 변화된 부분만을 저장하기 때문에 I 프레임에 비해서 상대적으로 크기가 작다. 또한 같은 종류의 프레임이라 하더라도 압축되는 영상의 성격에 따라 압축률이 달라지기 때문에 그 크기에 차이가 있을 수 있다. 일반적으로 P 프레임은 I 프레임의 약 3분의 1정도의 크기를 가지고 있으며, P 프레임과 B 프레임의 크기의 비율은 약 2:1에서 5:1 정도에 이른다.

그리고 동영상에서 한 장면으로부터 다음 장면으로 전환되는 시간은 일정하지 않으며, 하나의 장면전환으로부터 다음 장면전환까지의 장면들은 비슷한 성격을 가지게 된다. 만일 장면전환이 일어나게 되면 상호참조의 의미가 없어지기 때문에 P 또는 B 프레임이 있어야 할 시점이라도 다시 인트라 코딩이 된 프레임이 나타날 수 있다. 이러한 MPEG의 특징들은 실제 영상의 내용에 매우 의존적이기 때문에 프레임들의 크기를 예측할 수가 없다. 예를 들면 TV 드라마나 정적인 장면이 많은 예술영화에 비해 빠른 장면전환이 주를 이루는 CF나 중국 무협영화같은 영상들의 경우에는 데이터 양이 더욱 많아질 수 있다.

실제 MPEG 테이터를 통해서 MPEG 트래픽을 수학적으로 모델링한 Krunz [8]의 실험결과에 따르면 MPEG에서 I, B, P 프레임들의 크기는 각각 lognormal 분포와 유사하며 또한 한번의 장면전환에서 다음 장면 전환까지의 기간은 geometric분포와 유사하다. 이렇게 MPEG 테이터의 크기 변동은 영상 내용과 관련이 많기 때문에 오류 발생시 수신 측에서 어떠한 프레임에 오류가 발생했는지의 여부를 알기가 매우 힘들다.

3. 시스템 모델 및 가정

제안한 시스템은 기존의 멀티미디어 응용을 위한 전송시스템에 가상버퍼와 재전송 채널을 추가하였으며 재

전송 요구는 기존의 제어채널을 이용한다 ([그림 4]).

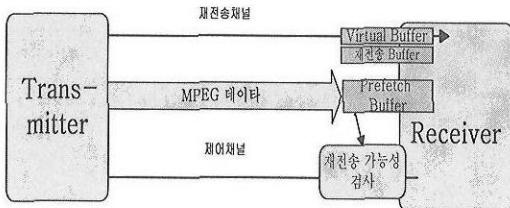


그림 4 제안한 시스템 구조

앞장에서 살펴본 것과 같이 MPEG은 GOP의 주기적인 반복으로 나타나지만 각각 프레임들의 크기를 예측할 수 없기 때문에 손실 받은 셀을 포함하는 프레임이 어떤 프레임인지, 어느 범위까지 그 영향이 전파되는지를 예측하기가 힘들다. 따라서, 본 논문에서는 MPEG 데이터 전송 시 예측성을 확보하는 방안으로써 가상버퍼 메카니즘을 도입하였으며 이를 이용하면 오류발생 시 선택적 재전송이 가능하게 된다. 가상버퍼는 MPEG 데이터 스트림의 프레임 순서, 종류, 크기 등에 대한 정보를 해당하는 프레임 재생시점 이전에 미리 가지고 있기 때문에 오류 발생 시점에서 버퍼내의 데이터 양과 가상버퍼를 비교하면 프레임의 종류 및 손실로 인한 영향이 미치는 범위를 파악할 수 있다 ([그림 5]).



그림 5 가상버퍼 구조

재전송 채널은 재전송 요구가 있을 경우 오류 데이터 부분만을 전송해주는 역할을 하며, 평상시에는 가상버퍼를 유지하기 위해서 향후 전송될 MPEG 데이터에 대한 간단한 정보들을 지속적으로 재생 측으로 전달한다. 가상버퍼 데이터와 재전송 데이터는 그 양이 상대적으로 작기 때문에 수 K-bytes/sec 정도의 대역 폭만으로도 충분하다. 재전송 요구를 송신 측으로 보내기 위해서는 일반적인 멀티미디어 응용에서 제어명령 (예를 들면, play, stop, fast forward, rewind 등)을 전달하기 위해 사용하는 기존의 협대역 제어채널을 사용한다. 전송중 데이터 오류가 발생한 경우에는 이 데이터가 수신측의 사전추출 버퍼 (prefetch buffer)에 들어가기 전에 검출하여 재전송 가능성 여부를 검사한다.

또한 본 논문에서는 MPEG 트래픽을 전송할 때

ATM을 사용한다고 가정한다. 일반적으로 ATM의 응용계층에서 가변비트율의 MPEG 트래픽을 전달하기 위해서 AAL5의 사용이 권장된다. 그러나 AAL5는 원래 비실시간의 성격을 지닌 데이터 트래픽을 전달하기 위해 설계되었으며 여러 개의 셀들로서 구성되는 PDU (Protocol Data Unit)레벨에서 CRC 검사 등을 통해서만 에러를 검출해 낼 수 있을 뿐 셀 레벨에서의 전송오류 또는 셀손실 등을 검출해 내지 못한다 [10][9]. 따라서 전송중에 셀손실이 발생했을 경우 그 하나의 셀 뿐만 아니라 그 셀을 포함하고 있는 PDU 전체를 폐기해야만 하기 때문에 2장에서 살펴본 MPEG의 셀손실에 의한 오류를 증폭시키는 경우가 발생할 수 있다. [1]는 이러한 문제를 해결하기 위해서 AAL5의 셀 구조에서 1바이트의 셀 순서번호 (cell sequence number)를 추가해서 셀 레벨에서의 손실을 발견할 수 있도록 AAL5를 개선한 MAAL (Multimedia AAL)을 제안하였다. MAAL은 셀 순서번호를 추가함으로서 그만큼의 오버헤드를 유발하지만 기존의 PDU 레벨에서 셀 레벨로 오류 검출의 해상도 (resolution)를 높여 오류에 대한 대처능력을 증가시키고 손실의 영향을 줄일 수 있다는 장점이 있다. 본 논문에서는 전송중의 셀 손실이 발생했을 경우 이의 재전송을 위해 손실된 셀을 검출해내야만 하기 때문에 이러한 MAAL 메커니즘을 통해 MPEG 트래픽을 전송한다고 가정한다.

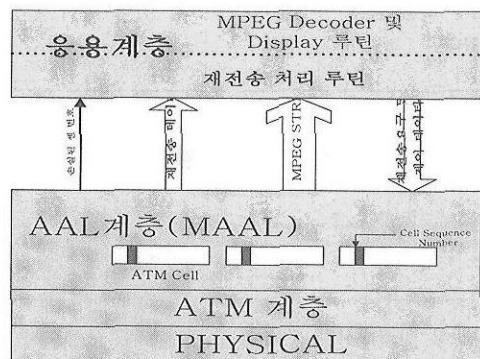


그림 6 수신측의 프로토콜 스택

위의 [그림 6]은 MAAL과 본 논문에서 가정한 재전송 루틴을 적용한 수신측의 프로토콜 스택의 구성도이다. 즉 ALL계층에서는 데이터 전송시에 셀손실이 발생했을 경우 손실된 셀의 위치정보를 응용계층으로 전달해 주며 응용계층의 재전송 처리 루틴에서는 AAL계층으로부터 받은 셀의 위치정보를 이용하여 재전송이 가

능할 경우에는 재전송을 요구한다.

한편 본 논문에서 가정은 다음과 같다.

- A1) 재전송 중에는 데이터의 오류가 발생하지 않는다.
- A2) 재전송 및 제어채널은 단 방향이다.
- A3) 송신 측은 수신 측에게 MPEG으로 인코딩된 데이터를 전송하며 이는 네트워크 어댑터로부터 버퍼에 저장되어 재생되기까지 최대 약 500ms 정도 머문다 (여기서, 500ms는 15개의 프레임으로 구성된 한 개의 GOP 재생시간을 의미하며 실제, 실시간 응용에서는 150~200ms이 적당하나 비실시간 응용에서는 500ms도 가능하다).
- A4) 수신 측에서 유지해야 하는 가상버퍼내의 최소 데이터의 수는 실제 버퍼에 최대한 저장할 수 있는 프레임의 개수에 한 개의 GOP의 프레임의 개수를 더한 만큼으로 한다.
- A5) MPEG 데이터전송은 AAL5를 수정한 MAAL [1]을 이용하며 재전송채널과 제어채널은 데이터의 변동폭이 적으므로 AAL1을 이용한다.

4. 가상버퍼를 이용한 선택적 재전송 기법

가상버퍼를 이용한 선택적 재전송을 수행하기 위한 절차는 다음과 같다. 먼저, 전송중 오류가 발생한 프레임이 있는지의 여부를 판단해야 한다. 이는 MAAL을 통해 전달되는 각각의 셀들에 순서번호 (sequence number)가 포함되어 전송되며 수신 측에서는 이 순서번호를 검사하여 셀의 손실 (cell loss) 또는 셀의 잘못된 삽입 (cell miss-insertion)이 발생하였을 경우 이를 상위의 응용계층으로 알려준다[1]. 때문에 전송중의 손실이나 잘못된 삽입 등으로 인해 발생하는 오류를 검출해 볼 수 있다.

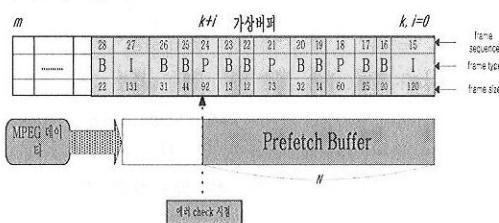


그림 7 가상버퍼를 이용한 오류발생 프레임 검출방법

그런 다음, 오류가 발생한 프레임을 찾기위해서 바로 다음 재생될 프레임에 대응하는 가상버퍼내의 프레임 ([그림 7]에서 k)으로부터 앞으로 재생되어질 프레임들의 크기를 차례로 더해 누적시킨 값을 오류발생을 검출해 낸 시점에서 실제버퍼에 있는 데이터의 양과 비교하는

방법을 사용한다. 이는 가상버퍼가 MPEG프레임들에 대한 프레임 순서, 프레임 크기 등의 기본적인 데이터만을 포함하고 있으나 실제 버퍼와 정확하게 매핑되기 때문에 위의 [그림 7]과 같이 예리가 발생한 시점의 실제 버퍼 크기를 가상버퍼와 비교하면 어떤 프레임이 오류가 발생한 프레임인지를 알 수 있게 된다. 이러한 조건을 수식으로 표현하면 다음과 같다.

$$\sum_{i=0}^m VB[k+i].frame_size > N \quad (1)$$

이때 VB 는 가상버퍼이며 k 는 다음 번에 재생되어질 프레임의 순서번호(frame sequence), m 은 현재 가상버퍼에서 유지하는 프레임의 개수, N 은 오류발생 시점에서의 사전추출 버퍼안의 데이터 스트림의 크기가 된다. 즉 i 를 증가시키며 위의 (1)식을 검사하여 이를 처음 만족시키는 $k+i$ 번째의 가상버퍼내의 프레임에 해당하는 실제 프레임이 오류가 발생한 프레임이 된다. 예를 들어 [그림 5]에서 곧 재생될 프레임 순서번호인 k 는 15이고, 오류가 발생한 시점에서 사전추출 버퍼 안의 데이터 양은 N 이다. 이때 15번 프레임의 크기 120을 16번째인 20, 17번째인 25 등의 순서로 더해 가면 24 번째의 프레임 순서번호에 이르러서 이의 누적된 값이 N 보다 크게된다. 따라서 프레임 순서번호 24 인 P 프레임이 오류가 발생한 프레임이다.

또한 오류에 의해 영향을 받는 프레임들을 찾는 방법은 가상버퍼로부터 오류가 발생한 프레임의 참조 관계에 따라 GOP의 끝을 찾아낸다. 예를 들어 [그림 7]의 경우에는 24번째의 P 프레임에서 오류가 발생하였기 때문에 이 오류가 뒤의 프레임들로 전파될 수 있으며, 이것의 영향은 GOP의 끝인 26번 프레임까지 미친다는 것을 알 수 있다. 이러한 가상버퍼로서 사용되는 데이터를 실시간으로 분리해 전달해 주는 것은 전송 측에서 많은 양의 버퍼와 처리시간을 요구하기 때문에 본 논문에서는 데이터의 인코딩 시 또는 사전작업을 통해 이를 독립적인 파일로 유지하는 것으로 가정하였다. 따라서, 추가적인 기억장치와 대역 폭이 필요하지만 방대한 크기의 멀티미디어 데이터에 비하면 그 크기는 상대적으로 작다.

오류가 발생한 프레임을 발견한 후에는 재전송을 통해 그 오류를 복구 해 줄 수 있는지의 여부를 검사한다. 일반적으로 네트워크를 고려한 멀티미디어 시스템에서는 네트워크 상의 지연과 지터(jitter)를 흡수하여 연속성을 보장해 주기 위해서 일정량의 사전추출 버퍼를 가지고 있기 때문에 [7] 실제로는 데이터가 도달한 순간

으로부터 재생되는 순간까지 어느 정도의 버퍼 지연(prefetching delay)을 가지게 된다. 그러므로 재전송을 요구해서 그 데이터가 돌아오는 시간이 버퍼지연보다 작다면 오류를 복구할 수 있다. 이를 식으로 표현하면 다음과 같다.

$$t_p(n) - t_a(n) \geq \text{Max Round Trip Delay} \quad (2)$$

위의 (2)식에서 $t_a(n)$ 은 n번째 프레임이 네트워크에서 버퍼에 도착하는 시간이며 $t_p(n)$ 은 n번째 프레임이 재생되어질 예측시간이다. 이때 $t_p(n)$ 은 가상버퍼를 이용하여 버퍼안에 들어가 있는 프레임의 개수를 파악함으로서 알 수 있다 그리고 최대 왕복지연은 재전송요구가 제어채널을 통해 송신자로 전달되어 송신자가 재전송 작업을 한 후에 재전송 채널을 통해 다시 돌아오기까지의 시간을 의미한다. 그러나 MPEG에서 손실된 데이터가 I 또는 P 프레임일 경우에 그 오류의 영향이 해당하는 GOP의 끝까지 전파되므로 이 GOP가 모두 재생되는 시간 이내에 손실 받은 데이터가 다시 도달한다면 그 이후 프레임들에 발생한 오류를 복구해 줄 수 있다. MPEG의 경우를 고려하면 재전송 여부를 판단하는 식은 다음과 같다.

$$t_p(n_{\text{last picture of GOP}}) - t_a(n) \geq \text{Max Round Trip Delay} \quad (3)$$

위의 (3)식에서 $t_p(n_{\text{last picture of GOP}})$ 은 n번째 프레임의 손실로 영향을 받는 마지막 프레임이 재생되어질 시간이다 따라서 오류가 발생한 프레임이 위의 식 (3)을 만족한다면 재전송을 통해 오류를 복구해 줄 수 있다.

ATM은 일반적으로 데이터의 재전송 기능이 없기 때문에 이러한 재전송 작업은 사용자 수준에서 해주게 된다. 그리고 재전송 회수를 줄이기 위한 방안으로서 오류가 자기 자신만으로 국한되는 B 프레임에 손실이 발생했을 경우에는 재전송을 수행하지 않는다 실제 B 프레임의 개수는 I 또는 P 프레임에 비해 많지만 프레임의 크기가 작기 때문에 오류가 발생할 확률이 상대적으로 작다.

이와 같이 본 논문에서 제안한 손실 데이터에 대한 선택적 재전송 알고리즘의 의사코드는 [그림 8]과 같으며, 이는 셀 손실이 발생했을 때 수행된다. 손실된 셀을 포함하고 있는 MPEG 프레임의 순서번호와 프레임 종류를 알기 위해서는 셀 손실 시 가상버퍼를 조사하여 각 프레임들의 크기를 더해가며 N 값과 비교해 가면 된다 (line 2-6). 손실된 셀을 포함하는 프레임이 B 프레임이라면 사용자에게 큰 영향을 미치지 않으므로 무시

하고 I 또는 P 프레임이라면 재전송 가능성을 검사한다 (line 8). 재전송 여부의 검사는 가장버퍼를 검사하여 현재 셀 손실에 의해 영향받는 마지막 프레임의 위치를 구한 후 (line 9) 이를 토대로 프레임의 재생시간을 구한다 (line 10). 이때 예상 재생시간을 얻는 방법은 재생될 때까지의 프레임의 개수를 구해 여기다 프레임 주기를 곱하면 된다. 그리고 이 오류에 영향받는 마지막 프레임의 예상 재생시간에서 현재 오류가 발생한 프레임의 도착시간을 빼서 이 값이 최대 재전송 시간(maximum round trip delay)보다 클 경우 재전송을 요구한다 (line 12-13).

```

1  when cell loss occurred
2      tot = 0; i=0;
         { find frame type and frame number
           influenced by cell loss }
3      while (tot < N) do
4          tot = tot + VB[k+i].frame_size;
         { k is next presented frame }
5          i = i + 1;
6      end while
7      n := k + i,   { cell loss occurred
                      at k+i th frame }
8      if ( VB[k+i].frame_type is I or P ) do
          { selective retransmission }
9          find last frame influenced by nth frame
          using virtual buffer: l;
10         calculate presentation time of tp(l);
11         calculate arrival time of cell loss
          frame n: ta(n);
12         if ( tp(l) - ta(n) > maximum round
              trip delay) do
13             request retransmission;
14         end if
15     end if

```

그림 8 오류가 발생한 프레임 검출 및 선택적 재전송 알고리즘

지금까지 설명한 선택적 재전송 기법을 요약하면 다음과 같다. 먼저, 오류가 발생한 프레임이 있는지 여부를 확인한 다음 오류가 발생한 프레임이 있다면 그것의 프레임 타입을 알아야 한다. 그런 다음 오류의 영향을 받는 프레임들의 범위를 검출한 뒤, 재전송 여부를 판단한다 만일 재전송이 필요하다고 판단되면 제어채널을 통하여 재전송 요구를 하고 송신 측에서는 재전송 채널을 이용하여 손실되었던 데이터를 재전송한다 그리고 마지막으로 오류 프레임을 복구한다.

본 논문에서 제안한 가상버퍼는 다른 분야에도 응용할 수 있다. 예를 든다면 서버에서 여러 개의 MPEG 스트리밍을 전송해 주어야 할 경우에 손실의 영향이 큰 I 프레임이 마감시간을 만족하지 못하거나, 프레임들이 연속적으로 마감시간을 보장하지 못하는 것을 막기 위해

긴급한 프레임들을 사전 할당하여 스케줄링 [4] 해 주는 경우에 가상버퍼로 사용되는 프레임의 크기정보를 사전할당을 위한 정보로서 사용할 수 있다. 또한 분산형 테이터의 멀티미디어 응용에서 하나 이상의 서버로부터 전달되어 오는 데이터들의 동기화 작업이 수행되어져야 할 경우에 수신측의 버퍼에서 동기화를 위한 동기화 마크의 역할도 수행할 수 있으며, 이밖에도 제한된 크기의 버퍼에 burstness가 큰 비디오 데이터들을 수신할 경우에 버퍼의 오버플로우를 막기 위해 중요도가 떨어지는 정보를 사전에 제거해 주기 위한 방안으로도 활용이 가능하다.

한편, 제안한 선택적 재전송 기법의 제약점으로는 가상버퍼를 위한 추가적인 기억용량과 대역폭이 필요하다는 것이다. 그러나 가상버퍼를 유지하기 위한 데이터는 그 양이 실제 비디오 데이터에 비해서 상대적으로 매우 작기 때문에 이로 인한 오버헤드는 크지 않다. 또한, 가상버퍼의 내용은 앞으로 발생할 데이터에 대한 사전정보이기 때문에 화상회의와 같은 인터액티브한 환경에서는 적용하기가 어려우며, 재전송을 위한 추가 데이터 채널이 필요하다는 제약이 있다. 그리고, 재전송이 필요한 멀티미디어 응용일지라도 데이터의 연속성(시간적인 제약)으로 인하여 재전송 비용은 비싼 편이다.

5. 시뮬레이션

본 논문의 시뮬레이션은 Pentium 프로세서를 사용하는 Windows95 환경에서 MS-Visual C++ 4.0의 32비트 콘솔 응용프로그램으로 수행하였으며 시뮬레이션을 위하여 MPEG 트래픽의 생성기를 제작하였다.

5.1 트래픽 생성기와 시뮬레이션 모델

MPEG 트래픽의 특성은 앞서 살펴본 것과 같이 I, B, P 프레임들간의 압축방식의 차이에 의해서 프레임당 크기가 다르며 같은 종류의 프레임이라 하더라도 압축되는 영상의 성격에 따라 압축률이 달라지기 때문에 크기에 차이가 있다. 또한 비슷한 장면이 일정시간 유지되기 때문에 인접한 프레임들간에 상호관계를 가지며 장면전환에 의해 다시 인트라 코딩이 된 프레임이 나타날 수 있기 때문에 그 트래픽의 분포를 예측할 수가 없다. 때문에 본 논문에서는 시뮬레이션을 위해서 Krunz [8]의 실험결과에 따른 [그림 9]에 나타난 의사코드를 이용해서 MPEG 트래픽 생성기를 제작하였다.

Krunz는 ‘오즈의 마법사라는 23분 정도의 영화를 소프트웨어 인코더를 통해 인코딩한 후에 I, B, P 프레임들 각각의 분포와 장면전환의 기간에 대한 분포를 수학적으로 분석하여 이의 모델을 설정하였다. 이에 따르면

```

1   set Scene_max := desired number of scenes
2   set values for compression pattern: L,Q
3   set scene length parameter: q
4   set the values for
5        $\tilde{\mu}_I, \tilde{\sigma}_I, \mu_P, \sigma_P, \mu_B, \sigma_B, a_1, a_2$ , and  $\sigma_\epsilon^2$ 
6   X = frame size(in cells)
7   for j=1 to Scene_max do
8       generate the length of scene j:  $N_j \sim \text{Geometric}(p)$ 
9       generate  $\tilde{X}_j(j) \sim \log normal(\tilde{\mu}_I, \tilde{\sigma}_I)$ 
10      for I=1 to  $N_j$  do
11          for k=1 to L do
12              if  $k=1$ , then
13                   $X = \tilde{X}_j(j) + \Delta_I(i)$ , where
14                   $\Delta_I(i) = \hat{a}_1 \Delta_I(i-1) + \hat{a}_2 \Delta_I(i-2) + \epsilon(i)$ 
15              else
16                  if remainder( $k/Q$ ) = 1, then
17                       $X \sim \log normal(\hat{\mu}_P, \hat{\sigma}_P)$ 
18                  else
19                       $X \sim \log normal(\hat{\mu}_B, \hat{\sigma}_B)$ 
20                  else if
21                  else for
22                  else for
23      else for

```

그림 9 MPEG트래픽 생성기의 의사코드

I, B, P의 각 프레임들은 lognormal 분포에 가까운 형태를 띠고, 프레임간의 상호관계의 유지간격은 geometric 분포와 유사한 형태를 가진다. 참고로 [그림 9]에서, 장면의 첫 번째 프레임이 I이지만 항상 GOP의 시작을 의미하는 것은 아니다 (GOP의 시작과 장면전환과는 무관하다). [그림 10]은 제작한 트래픽 발생기의 출력중 I 프레임들의 크기별 분포를 보이고 있는데 이는 lognormal 분포와 매우 유사함을 알 수 있다.

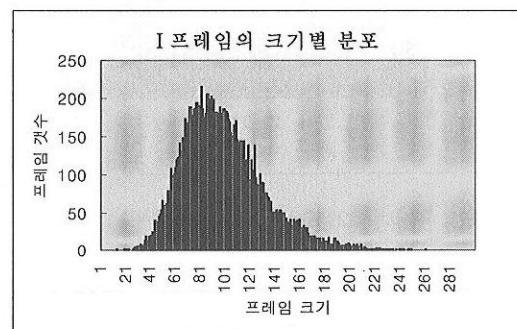


그림 10 I 프레임들의 크기별 분포

[그림 11]은 트래픽 생성기에 주어지는 초기 값으로서 전체 프레임들의 평균값과 표준편차, 그리고 lognormal 분포를 따르는 I, P, B 프레임들 각각의 평균과 표준편차이다. 또한 이때의 GOP는 15, M은 3으로

설정하였고 프레임간의 간격은 34ms를 가정하였다. 이 입력 값을 통해 발생한 유사 MPEG 트래픽은 평균 약 1.2Mbps정도의 용량을 가진다.

Statistic (Kbits)	All	I	P	B
Sample Mean	41.7	197.1	58.0	19.6
Sample STDV	51.7	63	37.3	5.7
Maximum	343.1	343.1	284.6	60.0
Minimum	0.56	36.2	0.56	0.57
GOP		15		
M		3		

그림 11 시뮬레이션 파라메터, 1.2Mbps (MPEG1)

[그림 12]는 트래픽 생성기에 [그림 11]의 파라메터를 입력으로 설정한 후에 2,000개의 프레임을 발생시킨 예로서 Krunz의 원래 트래픽과 유사하였다. 시뮬레이션은 [그림 12]와 같은 트래픽 패턴을 갖는 MPEG 테이터를 2시간, 즉 216,000 프레임을 재생하는 경우를 시험 하였으며 네트워크에서 오류의 발생 빈도는 10^{-6} , 왕복 지연은 450ms로 가정하였고 재전송 데이터는 그 양이 매우 적기 때문에 오류가 발생하지 않는다고 가정하였다. 이때 성능 비교를 위해 [그림 13]과 같은 4가지 경우에 대하여 각각 시뮬레이션을 수행하였으며 베피의

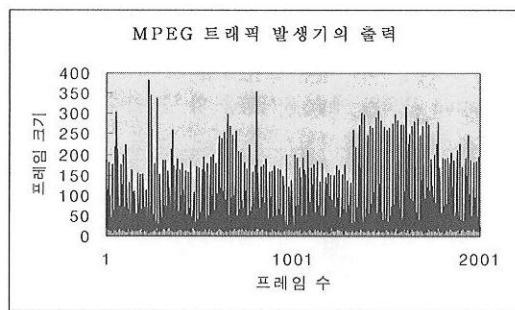


그림 12 트래픽 생성기의 출력

a	재전송 하지 않는 경우
b	MPEG의 특성을 고려하여 선택적인 재전송을 수행
c	단순히 현재의 프레임이 재전송되어 재생이 가능할 경우에만 재전송
d	에러가 발생하는 경우에 무조건 재전송

그림 13 시뮬레이션 대상

크기를 증가시켜가며 손실 받는 프레임의 개수, 재전송 회수, 3프레임 이상의 연속적인 손실이 발생한 프레임 수 등의 3가지를 비교하였다.

5.2 시뮬레이션 결과 분석

[그림 14]는 베피크기가 증가함에 따라 오류의 영향을 받는 프레임의 개수를 보이고 있다. 먼저 (a)의 경우는 재전송을 하지 않기 때문에 베피의 크기와 관계없이 언제나 일정한 수준의 손실을 보이고 있다. 그러나 (b)는 본 논문에서 제안한 베피의 크기와 왕복지연, GOP의 상태를 비교하여 선택적 재전송을 수행하는 방법으로서 베피의 크기가 증가함에 따라 재전송 가능한 프레임의 개수가 많아지고 이에 따라 손실되는 프레임의 개수가 감소하게 된다. 그러나 베피 크기가 15 (약 500ms) 이후에도 잊어버리는 프레임의 개수가 일정하게 유지되는 이유는 실제 사용자가 인지하기 힘든 B 프레임의 오류 시에는 재전송을 하지 않기 때문이다. (c)는 베피의 크기가 증가함에 따라 베피크기가 14에서 왕복지연이 베피의 크기보다 증가하게 되므로 재전송을 시작하여 오류의 영향을 받는 프레임이 0이 된다. (d)번은 무조건적인 재전송을 수행하므로 (b)의 경우와 같이 감소하다가 베피크기가 15 이후에는 0이 된다. 그러나 베피의 크기가 작을 경우에는 재전송을 하더라도 발생한 오류를 복구해 줄 수 없기 때문에 (a), (b), (d)의 3가지의 차이가 그리 크지 않다.

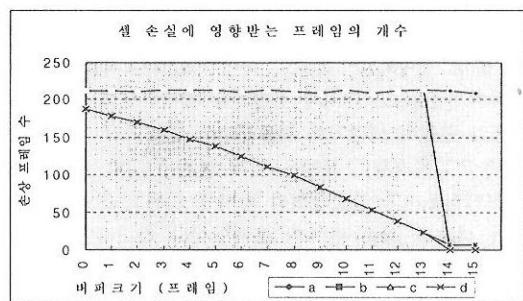


그림 14 셀 손실에 영향을 받는 프레임 개수

[그림 15]는 베피 증가에 따른 재전송 회수의 변화로서 (a)는 재전송을 하지 않는 방식이기 때문에 재전송 회수는 언제나 0이다. 본 논문에서 제안한 방법인 (b)는 오류의 영향을 받는 프레임의 개수와 반비례하여 재전송의 회수가 증가하게 된다. 이때 베피크기가 0에서 10까지 계단모양이 발생하는 이유는 GOP 내의 재전송 가능성 검사 시에 B 프레임의 오류는 무시되기 때문이다. (c)는 베피크기가 14에서 비로소 왕복지연이 베피의

크기보다 증가하게 되므로 재전송을 시작한다. (d)처럼 오류가 발생하는 경우에는 무조건 재전송을 하게 되므로 언제나 재전송 회수가 가장 크게 된다. 여기서도 역시 버퍼의 크기가 작을 경우에는 재전송을 통한 복구의 가능성이 작기 때문에 재전송 회수가 줄어들게 된다.

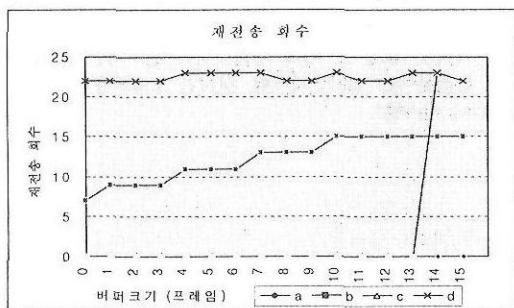


그림 15 재전송 횟수

[그림 16]은 전송 시 오류에 의한 영향이 3프레임 이상 지속되는 경우에 그 이후부터 손실되는 프레임의 개수를 구한 경우이다. 일반적으로 1 프레임만이 손실되는 경우에는 사람들이 이를 인지하지 못하기 때문에 실제 연속적인 손실의 경우만을 비교해 보았다. (a)에서는 재전송을 하지 않는 경우이기 때문에 손실의 회수가 일정하게 유지되며, 제안한 방법인 (b)는 무조건 재전송 하는 경우인 (d)의 경우와 동일한 값을 보이고 있다. 이때 재전송 회수의 증가가 계단현상을 보이는 것은 오류의 영향이 작은 B 프레임에 대해서는 재전송을 수행하지 않기 때문이다.

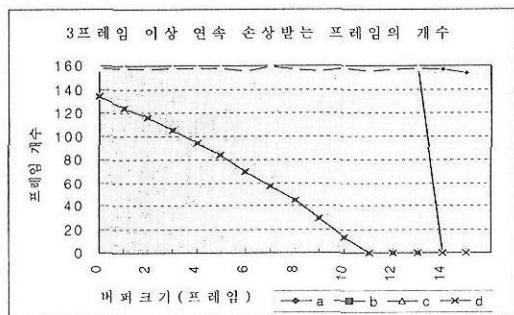


그림 16 3 프레임 이상 손상이 지속된 후에 손상된 프레임 개수

시뮬레이션 결과에 따르면 전송중에 오류가 발생했을 때 재전송 후에 MPEG 데이터의 오류복구에 기여할 수

있는 데이터를 선별하여 재전송 해주는 선택적인 재전송 기법이 적은 재전송 회수에도 불구하고 무조건 재전송을 수행하는 경우와 유사한 결과를 얻을 수 있음을 알 수 있다.

본 시뮬레이션에서는 약 1.2Mbps의 MPEG-1정도의 데이터를 이용하였기에 오류발생 프레임수 또는 재전송 회수가 적으나 앞으로 초고속 통신망 환경에서 수Mbps에서 수십Mbps이상의 고화질의 비디오를 전송할 경우에는 그 크기에 비례하여 오류 발생 빈도가 증가하게 되고 이에 따라 재전송 회수가 증가하게 될 것을 예상할 수 있다.

6. 결론 및 향후연구

본 논문에서는 MPEG를 사용하는 멀티미디어 응용에서 재전송이 필요한 경우 재전송 회수를 줄이면서도 QoS와 데이터의 연속성을 지원할 수 있는 선택적 재전송 기법을 제안하였다. 이는 다른 프레임을 참조하지 않고 독립적으로 재생이 가능한 I 프레임, 다른 I 프레임 또는 P 프레임을 참조해야만 재생이 가능한 P 프레임과 B 프레임으로 구성되어있는 MPEG의 특징으로 인해 전송중에 오류가 발생했을 경우 재생 시 이의 영향이 GOP 내에 전파될 수 있음을 고려한 것이다. 본 논문에서는 데이터 전송 시 발생한 오류의 전파범위를 파악한 뒤 재전송을 통해 그 오류를 복구해 줄 수 있을 경우에만 재전송을 수행하고, 오류가 발생한 프레임의 검출을 위해서 각 프레임들에 대한 기본적인 정보를 송신측에서 전송전에 미리 유지하는 가상버퍼를 도입하였다. 제안한 재전송 방법은 네트워크 지연이 GOP의 전송 시간 보다 크지 않은 환경에서 잘 작동할 수 있으며 화상의 일부 손실로도 QoS에 중대한 영향을 줄 수 있는 임무가 치명적인 멀티미디어 응용에 적합한 방법이다. 시뮬레이션 연구결과 제안한 재전송 기법을 사용하는 경우에는 데이터 전송중에 일부분의 데이터가 손상 받는 경우에도 적은 재전송 회수로 좋은 화질을 유지할 수 있음을 보였다.

한편, 본 논문에서 제안한 가상버퍼의 정보는 실시간 멀티미디어 데이터 전송 시 예측성을 부여하기 위한 방안으로서 스케줄링을 위한 사전할당 정보나, 분산환경에서 데이터 스트리밍 간의 동기화 정보로도 사용이 가능하며 이에 대한 추가연구를 수행할 예정이다.

참 고 문 헌

- [1] Xavier Garcia Adanez, Oliver Verschueren and Jean-Pierre Hubaux, Reliable Transmission of

- MPEG-2 VBR Video Streams over New Network Adaptation and ATM Adaption Layers
- [2] V. Bhaskaran, K. Konstantinides, *Image and Video Compression Standards Algorithms and Architecture*, Kluwer Academic Publishers, 1995
- [3] Miguel Correia, Paulo Pinto, "Low-Level Synchronization Algorithms on Broadband Networks", IEEE Proceeding of MULTIMEDIA'95, pp. 423-434, 1995.
- [4] Ching-Chin Han, Kang G. Shin, "Scheduling MPEG-Compressed Video Streams with Firm Deadline Constraints", IEEE Proceeding of Broadband Networks", IEEE Proceeding of MULTIMEDIA'95, pp.423-434, 1995.
- [5] L.C. Hu and K.J. Lin, "A Forward Error Control Scheme for Real-Time VBR Traffic in ATM", Proceedings of Pacific Rim International Symposium for Fault-Tolerant Systems, Newport Beach, Calif., Dec. 1995
- [6] J. Huang and P-J. Wan, "On Supporting Mission-Critical Multimedia Applications", Proceedings of the International Conference on Multimedia Computing and Systems, pp.46-53, Japan, 1996
- [7] Joseph Y. Hui, Ezhan Karasan, Jun Li, Junbiao Zhang, "Client-Server Synchronization and Buffering for Variable Rate Multimedia Retrievals", IEEE Journal on Selected Areas in Communications, Vol.14, No.1, pp.226-237 Jan. 1996.
- [8] Marwan Krunz, Ron Sass, Herman Hughes, "Statistical Characteristics and Multiplexing of MPEG Streams", Proceeding of INFOCOMM'95, pp.455-462, 1995.
- [9] David E. Mcdysan, Darren L.Spohn, *ATM Theory and Application*, McGraw-Hill, 1994.
- [10] Christos Tryfonas, *MPEG-2 Transport over ATM Networks*, University of California Santa Cruz Master of Computer Science, September 1996.



이 승 룡

1978년 2월 고려대학교 재료공학과 (공학사). 1986년 12월 Illinois Institute of Technology 전산학과 (석사). 1991년 12 월 Illinois Institute of Technology 전 산학과 (박사). 1992년 8월 ~ 1993년 8 월 Governors State University 조교수. 1993년 9월 ~ 1995년 10월 경희대학교 전자계산공학과 조 교수. 1995년 10월 ~ 현재 경희대학교 전자계산공학과 부 교수. 관심분야는 실시간시스템, 실시간 고장허용시스템, 멀티미디어 시스템.



이 석 훈

1995년 2월 경희대학교 전자계산공학과 (학사). 1997년 2월 경희대학교 전자계산 공학과 (석사). 1997년 2월 ~ 현재 (주) 교보정보통신 연구원. 관심분야는 실시간 시스템, 그룹웨어