

인터넷 전화에서 손실 패킷 복원을 위한 동적인 부가 정보 전송 기법

Dynamic Redundant Audio Transmission for Packet Loss Recovery in VoIP Systems

김 무 중^{*}, 권 철 홍^{**}
(Mu Jung Kim^{*}, Chul Hong Kwon^{**})

^{*}(주)언어과학 음성공학연구소, ^{**}대전대학교 컴퓨터·정보통신공학부
(접수일자: 2001년 9월 17일; 수정일자: 2002년 2월 15일; 채택일자: 2002년 5월 8일)

인터넷 폰 시스템은 네트워크 트래픽 문제로 인한 지연, 지터 그리고 패킷 손실을 경험하고 이로 인한 통화품질의 저하가 문제가 되어 통화품질 (QoS) 향상 기술이 필요하게 되었다. 본 논문에서는 인터넷상에서 통화 품질을 저해하는 요소들을 분석하고 실시간 전송 프로토콜/실시간 전송제어 프로토콜 (RTP/RTCP)을 이용하여 네트워크 상태를 진단하여 송, 수신 단말기간 네트워크 트래픽에 알맞은 방식으로 인코딩된 패킷을 송, 수신하는 동적인 손실 복구 알고리즘을 제안한다. 실험결과 제안한 부가정보를 이용한 동적인 손실 복구 알고리즘은 연속 패킷손실인 경우 63%의 손실패킷 복원률을 보여주며, 비연속 패킷손실인 경우 42%의 패킷손실 복원률을 보여준다.

핵심용어: 인터넷 전화, 통화품질, 실시간 전송 프로토콜/실시간 전송제어 프로토콜, 부가정보 전송

투고분야: 음성처리 분야 (2.1)

In ITU H.323 teleconference system, the RTP/RTCP protocol is offered to transfer real-time multimedia stream. Both sender and receiver have experience in packet loss and jitter which result from network congestion over Internet. Audio quality over Internet depends on the number of lost packets and on jitter between successive packets. The goal of our study is to improve the speech quality over Internet by checking the packet loss characteristics of the network and adopting the buffer control management mechanism at the receiver. We suggest a dynamic redundant audio transmission mechanism which examines the packet loss rate and uses the feedback information through RTCP.

Keywords: VoIP, QoS, RTP/RTCP, Redundant audio transmission

ASK subject classification: Speech signal processing (2.1)

I. 서론

오늘날 통신에 있어서 가장 중요한 관심사 중의 하나는 음성 데이터를 인터넷 프로토콜 (IP) 네트워크로 처리하는 인터넷 전화 (VoIP: Voice over IP)이다. 수년 전부터 여러 국내 통신기관과 기업체들이 이미 인터넷 전화 망을 구축하여 성공적으로 운영하고 있으며, 몇몇 국제 전화회

사들은 인터넷 전화를 이용한 국제전화 서비스를 개설하고 있기도 하며, 이러한 움직임은 인터넷 전화를 이용해 획기적으로 기업의 통신비용을 절감할 수 있다[1].

음성 데이터가 인터넷을 통해 전달되기 위해서 음성 신호의 디지털화, 압축, 그리고 패킷 형태로 전달하기 위한 프래그멘테이션, 인캡슐레이션 및 실시간 음성 서비스의 지원 그리고 사용자 위치 파악을 위한 네트워크 자원 설정과 기존 공중전화망에서 제공되는 부가서비스 지원 등의 기술이 요구된다[2].

음성 데이터는 송신측에서 수신측으로 전송시 지연에

책임저자: 권철홍 (chkwon@dju.ac.kr)
300-716 대전광역시 동구 용운동 96-3
대전대학교 컴퓨터·정보통신공학부
(전화: 042-280-2555; 팩스: 042-284-0109)

상당히 민감하다. 그러나 일반적으로 단방향 단-대-단 통신지연의 합계가 150 ms 이하이면 대부분의 이용자는 성능에 만족한다. 지연 이외에 수신 패킷간의 지연 편차인 지터 (jitter)가 발생한다. 지터는 전송되는 데이터 사이의 간격이 균일하지 않을 때 음성 품질을 현저히 떨어지게 한다. 전송시 패킷손실도 음성의 품질을 상당히 떨어지게 한다.

이와 같이 인터넷 폰 시스템은 네트워크 트래픽 문제로 인한 지연, 지터 그리고 패킷 손실을 경험하고, 이로 인한 통화품질의 저하가 문제가 되어 통화품질 향상 기술이 필요하게 되었다. 기존에 제안된 통화품질 향상 기법을 살펴보면[2], 인터리빙 기법을 이용한 시퀀스 재배열 방식은 송, 수신단에서 인터리빙 처리를 위한 프로세싱 타임이 많이 걸리며, 이산적인 패킷 손실시에는 복구율이 높으나 연속적인 패킷 손실시에는 패킷 복구율이 상당히 낮다. 순방향 에러 정정 (FEC: Forward Error Check) 방식은 송신단에서 에러 복구를 위한 부가정보를 보내는 방식으로 연속패킷복구를 위한 부가정보 및 이산패킷복구를 위한 부가정보 등을 송, 수신하기 때문에 패킷 손실시 복구율은 높으나 항상 부가정보를 송, 수신하기 때문에 네트워크의 과부하 및 단말기상에서 처리 과부하를 초래한다. 네트워크 트래픽 상태를 분석한 방법에서는 네트워크 트래픽 상태가 양호할 때는 주 정보만 송, 수신하지만 패킷 손실이 많은 네트워크 트래픽 발생시는 부가정보를 송신측에서 수신측으로 전송하기 때문에 동적으로 송, 수신할 수 있지만 부가정보의 데이터가 크기 때문에 많은 트래픽을 발생시킨다. 이와 같이 기존에 제안된 기법의 단점을 극복하기 위해서는 최적의 부가정보 유형의 선택, RTP/RTCP 헤더정보를 이용한 네트워크 트래픽에 따른 부가정보 전송여부를 정하는 기법이 요구되며, 따라서 본 논문에서는 RTP/RTCP 프로토콜을 이용하여 네트워크 상태를 진단하여 송, 수신 단말기간 네트워크 트래픽에 알맞은 방식으로 인코딩된 패킷을 송, 수신하는 동적인 손실 복구 알고리즘을 제안하고 그 성능을 분석한다.

II. 인터넷 전화 개요

2.1. H.323 구조

H.323은 멀티미디어 통신을 위하여 ITU (International Telecommunication Union)가 제정한 프로토콜이다. H.323은 하나의 시스템을 위해 사용되는 몇 가지 프로토콜의 조합으로 구성되어 있으며, 그림 1과 같은 프로토콜을 내

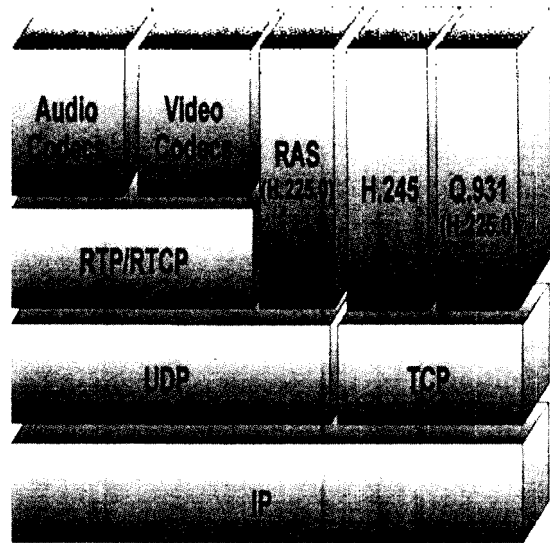


그림 1. H.323 구조
Fig. 1. H.323 structure.

장하고 있다. 시그널링 제어 등을 위해 H.245, H.225.0 등을 포함하고 있다. 오디오 코덱은 마이크에서 들어온 오디오 신호를 압축하는 역할을 한다. H.323에서 오디오는 기본적인 서비스이므로 H.323 단말기는 반드시 하나 이상의 오디오 코덱을 지원해야 한다. ITU에서는 G.711 (64 kbps), G.722 (64/56/48 kbps), G.723.1 (5.3/6.3 kbps), G.728 (16 kbps), G.729 (8 kbps) 등의 코덱을 지원할 것을 권장하고 있다. 비디오 코덱은 카메라를 통해 들어온 비디오 신호를 압축하는데 사용되며 비디오는 옵션으로 처리될 수 있으므로 H.323 단말기에서 반드시 지원하지 않지만, ITU에서는 비디오 전송을 위하여 H.261이나 H.263 같은 코덱을 지원할 것을 권장한다[2,3].

RTP/RTCP (Real-Time Transport Protocol / Real-Time Control Protocol)는 UDP/IP 전송 프로토콜의 단점인 비신뢰성을 보완하기 위한 프로토콜로서 각 미디어별로 두 개의 사용자 데이터그램 프로토콜 (UDP: User Datagram Protocol) 채널을 사용하는데 하나는 RTP, 다른 하나는 RTCP를 위해 사용한다. RTP/RTCP는 단방향 채널이다. 즉 한 채널을 통해 미디어를 동시에 송·수신할 수 없다[3].

2.2. 실시간 전송 프로토콜 (RTP)

오디오/비디오 데이터의 실시간 전송을 위해서 인터넷 표준화 기구인 IETF (Internet Engineering Task Force)에서는 RTP/RTCP에 관한 권고안을 발표하였다. RTP/RTCP는 대역폭을 예약하는 완전한 실시간 프로토콜은 아니지만, 현재의 UDP/IP 프로토콜에서 실시간을 지원할 수 있는 프로토콜로 사용되고 있다. 이러한 표준이 나오

게 된 이유는 일반적으로 TCP는 흐름제어 (flow control)와 재전송에 의한 패킷 손실 복구를 제공하므로 TCP는 실시간을 위한 응용에는 적합하지 않다. 그리고 UDP는 아무런 제어없이 패킷을 무조건 보내기만 하므로 UDP에는 다른 표준화된 상위 프로토콜이 필요하다[3,4].

RTP는 전송계층인 UDP/IP의 상위계층 프로토콜이다. RTP는 이름과는 달리 실시간 전송을 보장해 주지는 못한다. 다만 타임스탬프 (Timestamp)와 순서번호 (Sequence number)를 이용하여 패킷의 순서와 미디어간의 동기를 지원해 주게 된다. RTP는 UDP를 사용하기 때문에 패킷들의 손실이 발생할 수 있는데 RTP 내에서는 이러한 오류에 대한 복구기능을 지원하지 않는다. 따라서 이러한 오류가 발생할 때에는 오디오의 경우 하위 비트율 중복 (low-bit-rate redundancy)을 제공하거나 비디오와 같이 특별한 방법이 없는 경우 재전송 (retransmission)을 해야 한다[4].

2.3. 실시간 전송 제어 프로토콜 (RTCP)

RTCP는 특정한 RTP 세션에 대해서 참가자들에게 주기적인 제어패킷을 보내는 것을 기반으로 한다. 가장 기본적인 기능은 전송된 데이터의 품질에 대한 정보의 피드백을 제공하는 것이다. 이것은 전송 프로토콜로서 RTP의 역할에 대한 통합적인 부분이며 다른 프로토콜의 흐름 및 혼잡 제어와 관련이 되어 있다. 피드백은 적절한 인코딩/디코딩 방법 제어에 있어서 직접적으로 도움을 준다. 그리고 IP 전송 실험에 있어서 수신자로부터 피드백을 수신하여 에러율을 진단하는 것은 매우 중요한 부분이다. IP 다중 전송과 같이 데이터 분배 메카니즘에서는 세션에 참여하지 않고 있는 네트워크 서비스 제공자가 삼자 모니터로 피드백 정보를 이용하여 현재 네트워크 문제를 진단할 수 있다. 이러한 피드백 기능은 RTCP 송신자, 수신자 리포트들 통해서 수행된다[5].

III. 인터넷 전화에서 통화품질 제약 요소

연속적인 멀티미디어 전송을 하는 통신시스템은 데이터의 품질을 유지하고 동시에 요구되는 실시간 전송을 보장하여야 한다. 그러나 현존하는 인터넷의 속성을 살펴보면 송신측과 수신측이 회선을 점유하는 방식이 아니라 공유의 개념을 가지고 있어, 대역폭에 제한을 받는 패킷교환 방식으로 다수의 접속자들의 회선 공유로 인한 패킷의 손실 및 전송지연이 발생하여 음질을 떨어뜨린다[6].

음질에 영향을 미치는 요인으로 패킷손실, 전송지연, 지

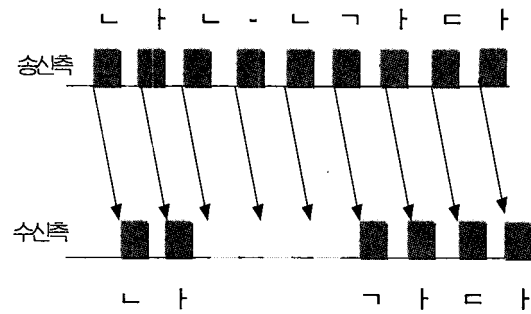


그림 2. 연속 패킷 손실
Fig. 2. Consecutive packet loss.

터 등이 있으며 이들에 대해 자세히 살펴보면 다음과 같다.

3.1. 패킷손실

음성신호는 시간축 상에서 상관도가 높기 때문에 일정 부분의 패킷손실에도 음성을 인식할 수 있다. 그리고 패킷손실이 일어났을 때 이전 프레임을 반복하거나 배경잡음을 삽입함으로써 어느 정도 패킷 손실에 의한 음성의 품질 저하를 막을 수 있다. 따라서 패킷 손실이 가끔 일어나는 경우 음성 품질에 영향이 별로 없다. 그러나 그림 2에서처럼 “나는 가다”라는 패킷을 전송시 연속적인 패킷손실이 발생할 경우 “는”의 음절단위의 패킷이 손실되어 인식할 수 없는 음질이 나타나므로 품질향상을 위한 알고리즘이 필요하다[7].

패킷 손실시 여러 복구 방법은 송신측 기반 복구 방법과 수신측 기반 복구방법으로 크게 나뉜다. 송신측 기반 복구 방법은 손실된 패킷을 복구하기 위해 송신측에서 부가정보 또는 손실된 패킷을 재전송하는 기법이 있으며, 수신측 기반 복구 방법은 손실된 패킷과 유사한 패킷을 생성하여 재배치하는 기술이다. 오디오 신호인 경우에는 인접한 패킷은 서로 유사한 특성을 가지고 있다. 그러므로 이러한 기술은 상대적으로 적은 손실률 (<15%)과 작은 패킷 (4 - 40 ms)에서 잘 동작한다[7].

3.2. 전송지연

지연에 관한 ITU 권고안 G.114에서는 만족할만한 음성 통신을 위해서 단방향 단-대-단 지연이 150 ms 이하가 되도록 요구하고 있다. 이 기준을 만족할 때에 기존의 회선교환 방식의 전화와 큰 차이가 없는 만족할만한 통화를 할 수 있다[8].

단-대-단간의 지연 요소는 고정 지연과 가변 지연으로 나눌 수 있다. 고정지연은 그림 3에서처럼 망 환경이 결정되면 고정이 되는 지연으로 전파 지연 (propagation delay),

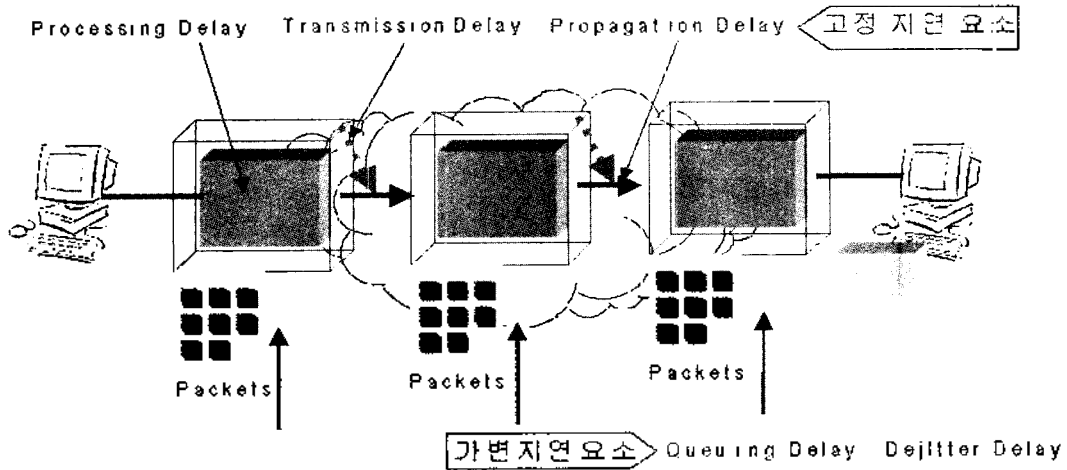


그림 3. 지연 요소
Fig. 3. Delay factors.

전송 지연 (transmission delay), 처리 지연 (processing delay) 등이 있다. 처리 지연은 다시 인코딩/디코딩 관련 지연, 패킷화 (packetization) 지연으로 나눌 수 있다.

가변지연은 그림 3에서처럼 디지털 버퍼 지연, 큐잉 지연 (queueing delay)이 있다. 디지털 버퍼는 가변지연을 고정지연으로 바꾸기 위한 것으로 모든 가변지연을 상쇄할만한 충분한 크기를 가지고 있어야 한다

위에서 설명한 고정지연과 가변지연을 모두 고려하여 단-대-단 간의 지연 버짓 (delay budget)을 계산하여 이 값이 150 ms 이하가 되어야 한다.

3.3. 지터

지터는 패킷간의 지연편차로 패킷 지터는 인터넷 전화에서 음성 품질에 중요한 영향을 미치는데, 그 이유는 음성 패킷이 재생될 때 일정한 시간 간격마다 재생되어야 하기 때문이다. 코덱에서 음성 패킷을 만들 때에는 일정한 주기로 만들어서 전송하게 된다. 그러나 망을 통해 패킷이 전달되면서 각 패킷들 사이의 지연이 일정하지 않게 된다. 경우에 따라서는 순서가 뒤바뀌는 수도 있다. 따라서 이러한 현상을 없애기 위해서는 수신측에서 지터를 제거하는 디지털 버퍼를 두게 된다. 디지털 버퍼의 크기는 지연 버짓에서 가변지연을 모두 합친 것에 약간의 여유를 둔 것이다.

IV. 동적 부가정보 전송 기법

4.1. 인터넷 폰 시스템 구현

본 연구에서 구현한 인터넷폰 시스템의 구성도는 그림

4와 같고, 테스트 환경은 인터넷상에서 11 hop 거리의 두 단말기로 구현한 인터넷 폰 시스템으로 테스트를 하였다.

구현된 인터넷 폰 시스템은 윈도우 시스템 기반에서 MCI (Media Control Interface) 프로그래밍 기법으로 작성하였으며, 음성 코덱은 G.723.1 코덱을 사용하였으며, 단-대-단 통신을 위해서는 WinSock 2를 이용하여 UDP 프로토콜 소켓 프로그래밍을 하였다[9].

본 시스템에서는 G.723.1 코덱에서 인코딩 입력으로 받을 수 있는 음성 블록 형식인 8KHz 샘플링, 모노, 16 bit로 구간을 30 ms씩 녹음하여 음성 프레임을 생성하고, 녹음 데이터 저장을 위한 480 바이트의 메모리를 연속적인 녹음을 위해서 5개를 할당하고 연결 형태의 큐 (queue)로 설정 관리하였다. 녹음 장치는 다섯 개 블록에 대하여 녹음이 끝나면 응용 프로그램은 G.723.1 코덱으로 480 바이트 음성데이터를 전달한다. 코덱으로 전달이 끝난 빈 블록은 신규 입력 음성을 저장할 수 있도록 재사용하였다. 재생부에서는 코덱을 통하여 디코딩된 오디오 정보를 저장할 메모리를 응용 프로그램으로부터 할당받은 후 응용 프로그램은 음성 데이터를 재생장치에게 전달하고, 재생장치는 음성 데이터 블록을 재생하게 한다. 응용 프로그램은 재생이 끝난 메모리 블록을 네트워크 수신측에서 다시 사용할 수 있도록 유지하기만 하면 된다. 재생용 메모리 블록은 네트워크 수신측에서 재사용되어 녹음용 메모리 블록과 마찬가지로 링버퍼의 형태를 갖추게 된다.

입력된 음성 샘플은 G.723.1 코덱을 통하여 압축된 음성을 생성한다. G.723.1의 특성을 살펴보면 음성 검출 (VAD: Voice Activity Detection) 알고리즘과 묵음 기간 동안 인공적인 배경잡음을 생성하기 위한 잡음 생성 (CNG: Comfort Noise Generator) 알고리즘을 이용한 묵음 압축

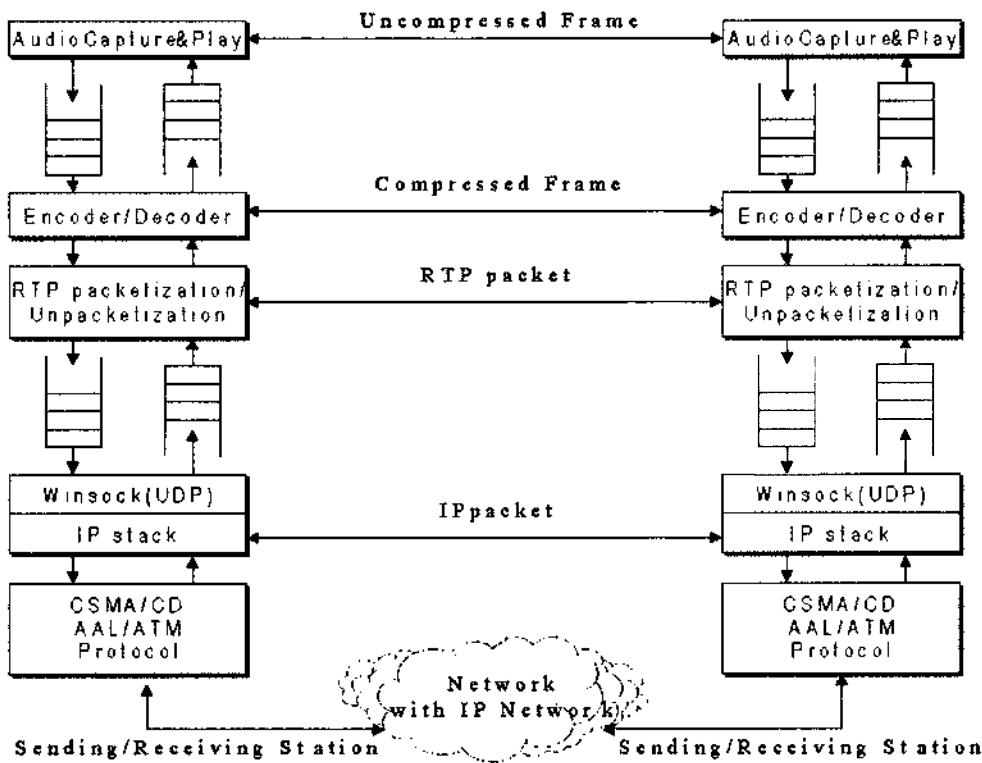


그림 4. 인터넷 폰 시스템 구성도
Fig. 4. Block diagram of the Internet phone system.

기법을 사용하여, 240 샘플의 음성 데이터는 G.723.1에 의해 인코딩되어 24 바이트 인코딩 음성과 4 바이트 묵음 검출(SID: Silence Insertion Descriptor) 신호, 그리고 1 바이트의 묵음프레임이 생성되며, 이중 24 바이트 인코딩 음성과 4 바이트 묵음 검출 신호는 UDP 프로토콜로 수신측에 전송이 되며, 1 바이트 묵음프레임은 전송이 되지 않는다. 또한 수신측에서는 24 바이트 인코딩 음성을 수신받아 디코딩 과정을 거쳐 240 샘플의 음성 데이터를 생성하고 4 바이트 묵음 검출 프레임 입력받으면 잡음 생성 알고리즘을 통해 묵음을 생성해 낸다[10].

실시간을 요구하는 멀티미디어 데이터는 IP 프로토콜 상에서 RTP 헤더를 UDP의 상위 계층에서 인캡슐레이션 후 전송하게 된다. UDP 프로토콜을 통해 전송시 특정 포트를 사용자가 지정 전송하여 빠른 연결설정을 할 수 있도록 구현하였다.

송수신시 RTP 헤더의 정보를 검사하여 통화품질 테이블의 타임스탬프와 순서번호 파라미터 값을 정하게 되고, 이들을 통하여 전송 지연, 지터 및 패킷손실 등을 계산하여 RTCP를 통해 피드백 정보를 송신측에 전송하게 된다.

4.2. 통화품질 파라미터의 계산

4.2.1. 패킷손실

수신측에서 수신한 RTP 패킷 분석을 통한 패킷손실 특성은 먼저 수신한 패킷 순서번호(seq)와 수신측의 통화품질 테이블에서 전 패킷의 순서번호로 계산된 last_seq 값으로 패킷 손실 수(count_loss)를 구할 수 있다.

$$\text{count_loss} = \text{seq} - \text{last_seq} \quad (1)$$

식 (1)에서 구한 count_loss가 4 이상인 경우 연속적인 패킷손실에 해당한다.

총 수신 패킷 수(total_rcvd)는 손실 패킷 수와 수신한 패킷 수를 더하여 구하고 식 (2)에서와 같이 패킷손실률(loss_ratio)은 총 패킷손실 수를 총 수신패킷수로 나누어 계산한다.

$$\text{loss_rate} = \text{count_loss}/\text{total_rcvd} \times 100\% \quad (2)$$

비순차적 패킷수신일 경우는 패킷손실로 분류한다.

4.2.2. 전송 지연

수신한 패킷간의 시간 간격인 전송 지연(delay)은 RTP 패킷헤더 정보에서 타임스탬프 값으로 계산하게 되며 식

(3)과 같다. 수신한 패킷 타임스탬프 (ts)에서 최근에 수신한 패킷의 타임스탬프 값 (last_pakktime)을 빼면 된다.

$$\text{delay} = \text{ts} - \text{last_pakktime} \quad (3)$$

4.2.3. 지터

패킷간의 지연편차인 지터는 계산된 지연의 증·감으로 파악할 수 있으며, 송신측 패킷의 타임스탬프 값에서 수신측 타임스탬프 값을 빼서 3가지의 특성으로 나눌 수 있다. 첫 번째 경우는 지연이 동일한 경우이고, 두 번째 경우는 지터가 발생한 경우이며, 세 번째 경우는 버스트 (burst) 패킷이 수신된 경우로 구분된 특성으로 지터의 증·감을 표시할 수 있다.

4.3. 동적 부가정보 전송에 의한 손실 복구 기법

네트워크 상의 패킷손실 특성 (패킷 손실률, 패킷손실 연속성) 분석을 하여 패킷손실 형태에 따라 다음 두 가지 방식

의 손실 복구 메카니즘을 제안한다. 첫째, 패킷손실률이 임계값 15%를 초과하는가 여부에 따라 송신측의 패킷 손실 복구 메카니즘이 주 음성 패킷만을 전송할 것인가, 또는 부가 전송 메카니즘을 적용하여 부가정보를 동시에 전송할 것인가를 결정한다. 둘째, 연속 패킷손실 수 (consecutive packet loss number)에 따라 1-type 부가 전송을 할 것인가, 2-type 부가전송을 할 것인가를 결정한다.

4.3.1. 패킷손실 특성

본 시험의 측정 환경은 송신측, 수신측간에는 11개 노드를 경유하며 5일간 각 5회, 총 25회 측정하여 평균값을 도출하였다.

그림 5는 지연이 30 ms 이상 발생하면 패킷 손실로 처리한 경우의 패킷 손실률을 나타내며 14:00부터 16:00 사이에 최대 20% 정도의 패킷 손실률을 보인다. 그림 6은 패킷손실이 가장 많이 발생하는 하는 14:00부터 16:00까지 매 5분마다 200개 패킷을 12초 동안 송신하여 수신측에서

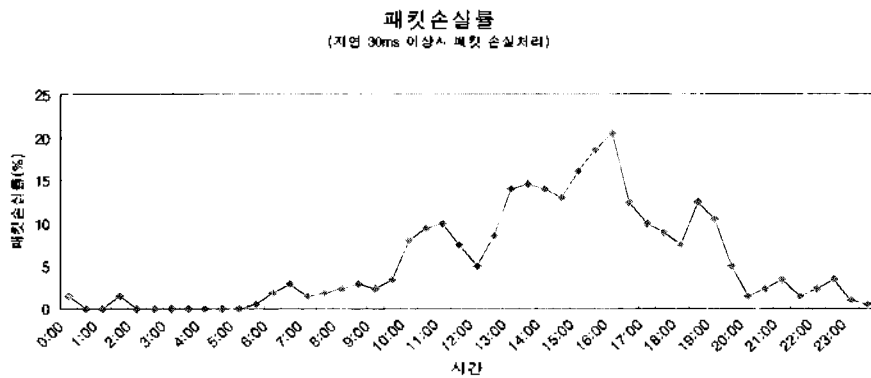


그림 5. 패킷 손실률 (0:00 - 24:00)
Fig. 5. Packet loss rate (0:00 - 24:00).

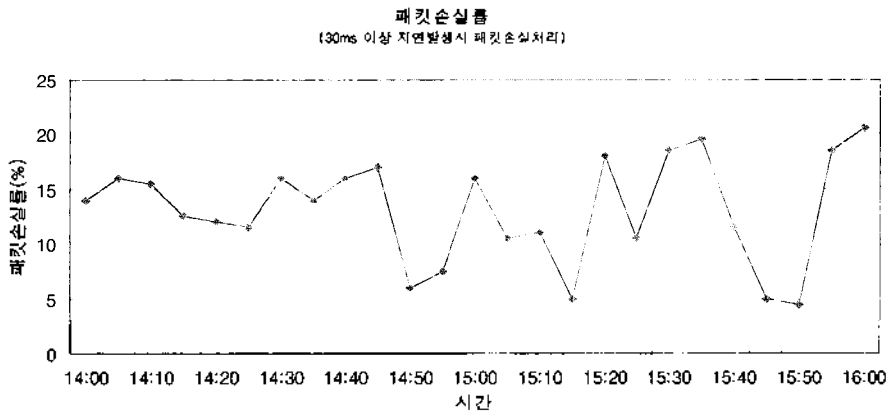


그림 6. 패킷 손실률 (14:00 - 16:00)
Fig. 6. Packet loss rate (14:00 - 16:00).

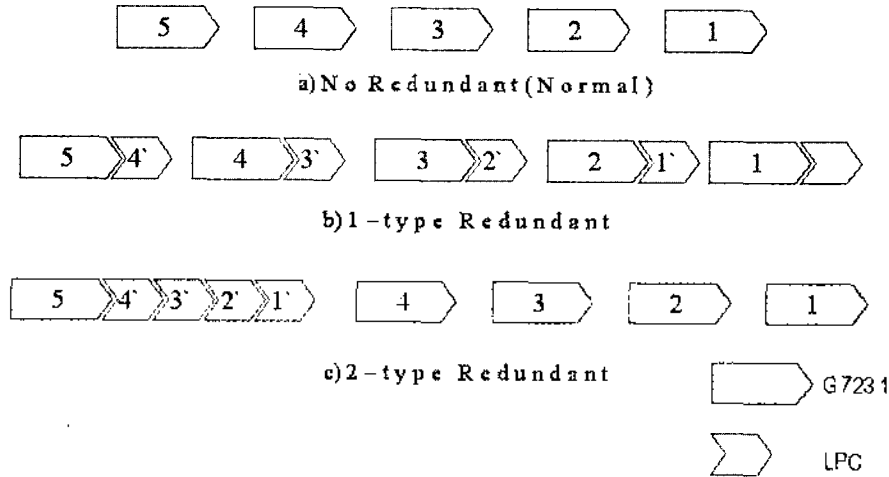


그림 7. 피드백 정보에 따른 부가 정보 전송형태
 Fig. 7. Classification of redundant audio transmission according to feedback information.

지연 30 ms가 발생시 패킷손실로 처리하여 계산된 패킷 손실률을 나타내며 최고 20.5%의 패킷손실이 발생하는 것을 볼 수 있다.

4.3.2. 수신측의 피드백 정보 전송에 따른 송신측의 부가정보 전송방식

부가정보 전송방식은 패킷손실을 고려하여 수신측에서 복구할 수 있도록 송신측에서 패킷의 중복성을 허용하는 방식으로 이를 위한 RTP 패킷은 주음성 정보와 패킷 손실을 대비한 부가음성 정보로 구성된다. 패킷 손실률이 낮은 통신환경에서의 부가 전송 메카니즘의 적용은 부가음성 정보 크기만큼의 대역폭 낭비를 초래한다. 따라서 송신측에서 패킷손실에 대한 정보를 알 수 있고, 부가 전송 메카니즘을 동적으로 적용한다면 부가 데이터에 해당하는 대역폭 낭비를 줄일 수 있다. 이러한 패킷손실에 대한 피드백 정보는 RTCP 프로토콜을 이용하여 수신측과 송신측이 정보를 교환할 수 있다. 예를 들어 임의의 패킷 손실률 임계값 ($\lambda=15\%$)보다 낮은 통신망 환경에서 부가 전송 방식은 패킷손실 복구에 대한 효율이 낮으므로 부가 전송없이 주음성 정보만을 전송함으로써 불필요한 대역폭 낭비를 감소시킬 수 있다. 그러나 피드백된 패킷 손실 정보가 손실률 임계값 (λ)보다 클 경우 통신망이 많은 부하가 걸린 상태가 되므로, 부가 전송 방식을 적용하면 손실된 패킷을 수신측에서 복구할 수 있다. 이때 고려해야 할 점은 부가음성 정보를 어떠한 순서형태로 전송하느냐가 중요하다. 이는 패킷 손실에 대한 연속성에 의존한다. 이를 위해 각 수신측으로부터 패킷손실 연속수 (CPLN: consecutive packet loss number)를 RTCP를 통

해 수신측에 전달한다. 패킷손실 연속수가 4 이상인 경우, 계속적으로 발생하는 빈도율이 높으므로 부가전송 메카니즘을 적용할 경우 수신측에서 복구 메카니즘으로 손실된 패킷을 복구 할 수 있다.

RTCP에 의해 피드백되는 패킷 손실 특성에 따라 그림 7에서와 같이 패킷 손실률이 15% 이상일 경우 1-type 부가형태로 전송하고, 연속손실 패킷수가 4이상일 경우 2-type 부가형태로 전송하는 2가지 모드로 구분하여 동적으로 전송방식을 선택한다.

그림 8의 패킷 손실 분류 알고리즘을 살펴보면, 음성 데이터의 패킷 손실률이 임계값 ($\lambda < 15\%$)보다 작은 경우

```

if PLR < λ
    then Site(i) = R(0)
else if PLR > λ AND CPLN ≤ μ
    then Site(i) = R(1)
else if PLR > λ AND CPLN > μ
    then Site(i) = R(2)

( PLR : Packet Loss Rate
  CPLN : Consecutive Packet Loss Number
  λ = 15%, μ = 4
  R(0) : no redundant state
  R(1) : 1 - type redundant state
  R(2) : 2 - type redundant state )
    
```

그림 8. 패킷손실 분류 알고리즘
 Fig. 8. Block diagram of the packet loss classification.

트래픽이 평이한 상태에서 부가음성 정보없이 주음성 정보만 전송함으로써 부가전송에 따른 오버헤드를 최소화한다. 임계값 ($\lambda \geq 15\%$)보다 큰 경우, 패킷 손실의 연속성에 따라 부가정보 전송 모드로 전환하여 송신측이 부가정보 전송 형태 중 하나의 형태로 전송할 수 있게 피드백 정보를 RTCP 프로토콜을 통해서 전송하게 된다.

부가정보 전송 방식에 의해 수신된 RTP 패킷의 수신측 처리는 그림 9의 순서도에서와 같이 RTP 패킷의 순서번호의 검색에 의해 손실 상태를 관찰하고, RTP 헤더의 확장 필드에 의해 부가정보 전송 여부 및 전송 형태를 확인할 수 있다.

코덱 모듈이 손실된 음성데이터 재생을 위해 기다리고 있지 않고 버퍼에 재생할 데이터들을 이미 갖고 있다면, 시스템 처리와 코덱 모듈 처리가 비동기적이므로 이러한 지연 시간은 재생에 영향을 미치지 않을 수 있다. 여기서 부가음성 정보 블록의 디코딩 처리시간은 유형에 따라 다르지만 G.723.1은 약 30 ms 정도의 시간이 소요된다. 또한 재생 시점은 음성 재생을 위해 코덱 디바이스 드라이버를 통해 버퍼에 음성 데이터를 전달한 시점으로 부가음성 정보 디코딩에 의해 소요되는 시간은 코덱 버퍼에

의해서 후속 데이터들의 재생 시작 시간에 영향을 주지 않는다.

4.4. 부가정보 전송 방식을 이용한 RTP/RTCP 구현 모델

송신측은 RTP 헤더에 통화품질을 위한 파라미터로 음성 데이터를 샘플링한 시점의 타임스탬프와 패킷 발생 순서를 지정하고 수신측은 이를 이용하여 패킷 손실, 전송 지연, 지터 등의 실시간 데이터 전송 특성들을 알 수 있으며 이에 대한 패킷 손실 보상 메카니즘이나 미디어간 동기화 등의 기능을 지원할 수 있다. RTCP는 통신하는 송신측과 수신측에게 RTP와 다른 UDP 포트를 사용하여 비주기적으로 수신측에서 계산된 패킷손실 정보와 네트워크 지연에 대한 제어 정보 패킷을 전송함으로써 송신측과 수신측간에 검출한 네트워크의 통화품질 정보를 전달하는 기능을 갖는다. 그림 10은 부가정보 전송방식을 이용한 RTP/RTCP 구현 모델을 나타낸 것으로 수신측은 통화품질 모니터를 통하여 패킷헤더를 분석하고 송신측에게 피드백 정보를 RTCP 프로토콜로 전송하며, 송신측은 수신한 피드백 정보를 기반으로 부가정보 형태를 결정하여 보낸다.

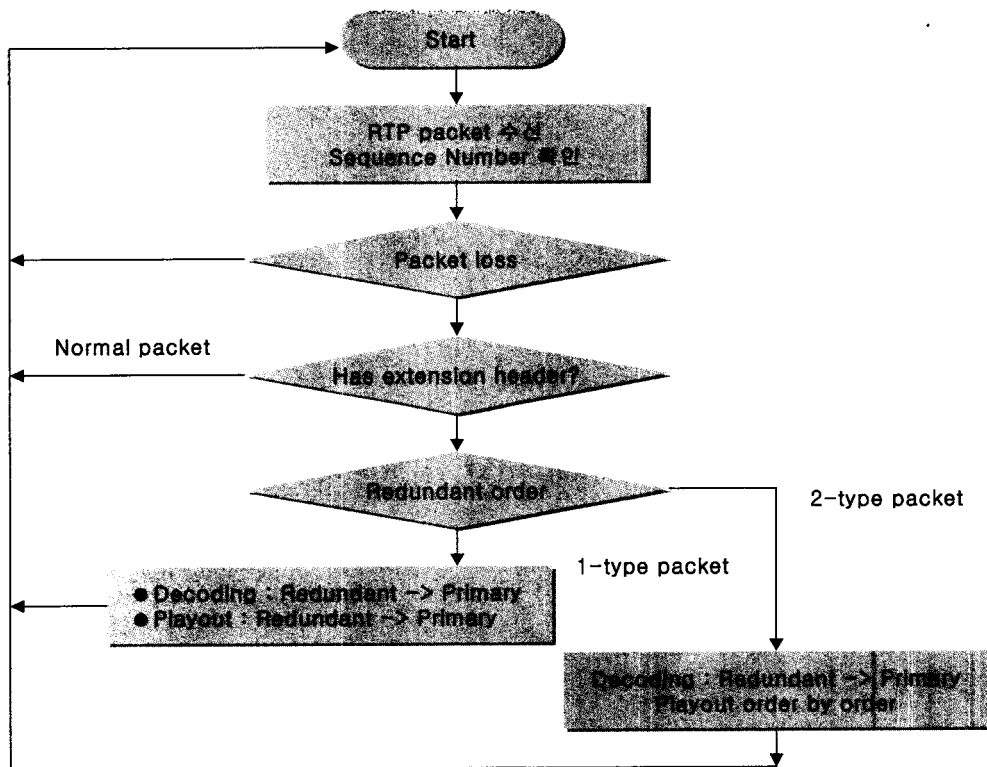


그림 9. 수신측 패킷 복구 알고리즘

Fig. 9. Block diagram of packet loss recovery at the receiver.

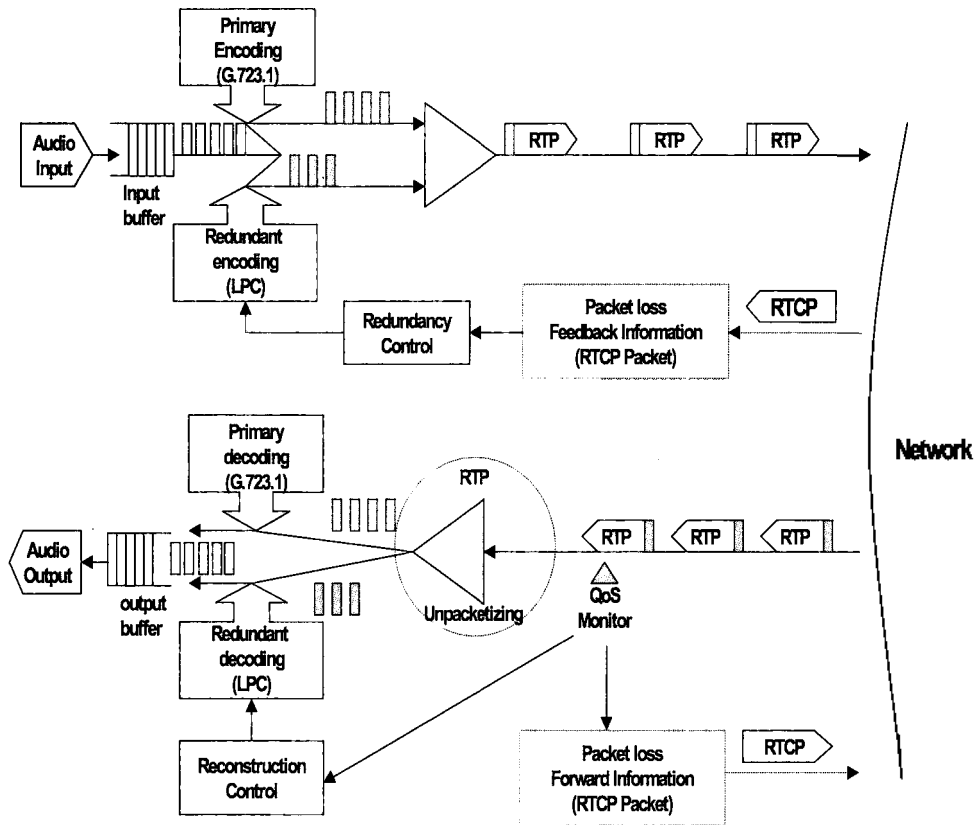


그림 10. RTP/RTCP 구현 모델
Fig. 10. Block diagram of the RTP/RTCP Implementation.

V. 시스템 평가 및 성능분석

인터넷상에서 음성 전송에서 나타난 결과에 대한 분석을 통하여 제안한 알고리즘에 대한 성능을 평가한다. 성능 평가를 위해 인터넷 폰 시스템의 RTP와 RTCP 프로토콜과 코덱을 이용한 패킷손실을 측정하였다. 실험환경은 교내 실험실 (203,237,141,212)과 11 hop 거리에 존재하

는 PC (211,240,96,85)와의 연결상태에서 통화 실험을 수행했으며 통화품질 파라미터의 값은 RTP/RTCP 정보를 통해 추출하였다.

인터넷은 분산 멀티미디어 시스템에서 실시간 속성을 지원하지 않기 때문에 UDP/IP 프로토콜 상에서 RTP 프로토콜을 사용한 비신뢰적인 통신을 한다. 그러므로 RTP/RTCP 프로토콜을 포함한 인터넷 폰 시스템은 패킷 손실을

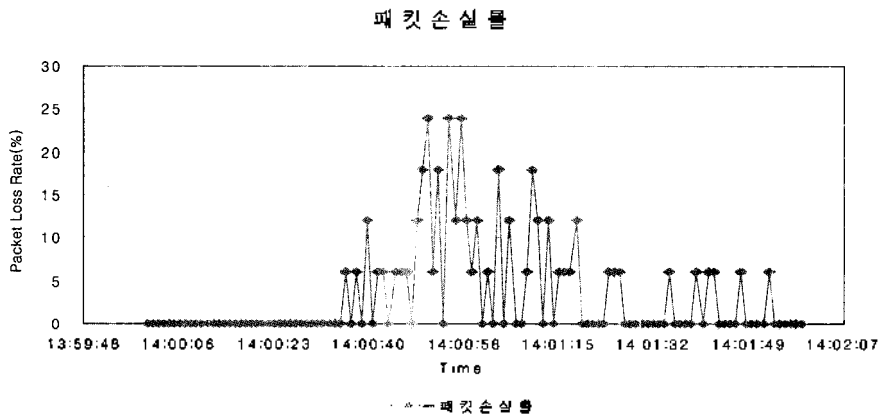


그림 11. 패킷 손실률 (14:00-14:02)
Fig. 11. Packet loss rate (14:00-14:02).

가져올 수 있으며 패킷 손실은 음성 품질의 저하를 가져오게 된다. 그림 11과 12에서 지연이 많이 발생하는 14:00 - 14:02 간의 교내 실험실 내와 외부망과 연결된 단말기 상에서 패킷 손실률 및 연속 손실 패킷 수를 나타내고 있다. 그림에 나타난 20% 넘는 패킷 손실은 음성 품질을 떨어뜨리는 경향이 있으며, 특히 3개 이상의 연속적인 패킷 손실은 90 ms 이상의 음성구간을 수신하지 못하였기 때문에 수신한 음성을 인식할 수 없는 현상을 초래한다.

그림 13은 제안된 동적 부가 전송 방식 중 1-type 부가 전송 방식만으로 손실 패킷을 복구한 경우 나타나는 손실의 연속성을 도식한 것이다. 이 그림에서 전체 패킷 손실 중에 비연속성 패킷 손실은 42%가 후속 패킷의 부가 정보 필드에 의해 복구되고, 연속성 패킷 손실의 경우 연속 패킷의 마지막 패킷이 복구되는 것을 볼 수 있다. 이때 고려해야 할 부가 전송 방식의 문제점으로 수신측에서 순서에 따라 도착하는 현재의 RTP 패킷을 수신하고 이전 패킷의

연속적인 패킷 손실 수

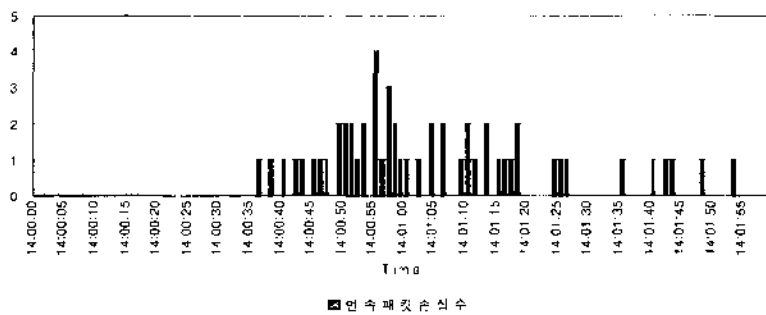


그림 12. 연속 손실 패킷 수 (14:00-14:02)
Fig. 12. Consecutive packet loss number (14:00-14:02).

1-type 부가정보 전송시 패킷손실수

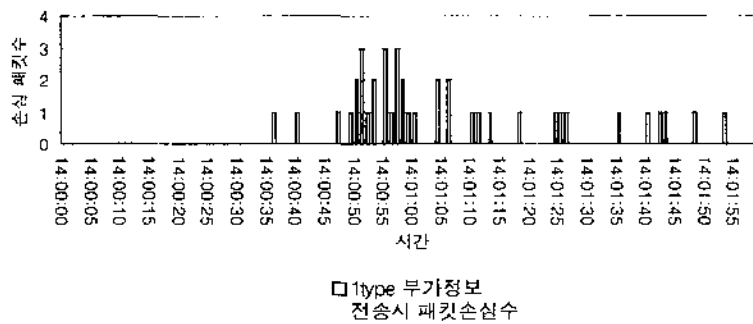


그림 13. 1-type 부가정보 전송에 의해 복원된 패킷 수
Fig. 13. The number of packets recovered by 1-type redundant audio transmission.

패킷 손실 복구 후 패킷 손실 수

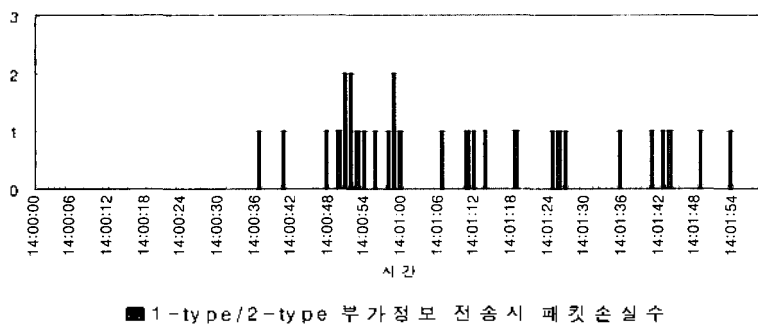


그림 14. 패킷손실 복구를 통한 패킷 손실 수
Fig. 14. The number of packets recovered by redundant audio transmission.

손실을 확인한 후, 현재 패킷에 부가 전송된 부가 정보 필드의 처리 시간에 대한 지연이 요구된다. 따라서 수신측에서 패킷 손실이 발생한 순간 해당 RTP의 부가 정보 필드를 분리하여 디코딩 처리하고 코덱에 전달하는 동안 해당 패킷의 주 정보 필드의 재생을 보류해야 한다. 이러한 보류 시간은 부가정보 필드의 디코딩 방식에 의존한다.

그림 14에서 연속적인 4개의 패킷이 발생 후 손실된 패킷의 복구는 부가정보 형태인 2-type에 의해 복구되며, 처음 발생하였을 경우에는 2-type의 부가 정보가 수신측에 수신되지 않았기 때문에 1-type에 의해 손실 패킷 중 한 개만 복구되나, 송신측에서 2-type의 부가 정보를 송신측에 송신시에는 연속적인 패킷이 발생하여도 3개 이상 복구되어 63% 정도의 패킷복구율을 볼 수 있다. 복구 시 복구에 필요한 지연이 발생되나 사람이 인지할 수 있는 허용범위 내에서 복구를 하기 때문에 인지하지 못한다.

VI. 결론

본 논문에서는 인터넷상에서 윈도우 운영체제를 기반으로 두 단말기간에 음성을 전송하는 인터넷 폰 시스템을 구현하였다. 실시간 전송을 위하여 RTP/RTCP 프로토콜을 사용하였다. RTP/RTCP 프로토콜은 음성 패킷의 실시간 전송을 보장해 주는 프로토콜이 아니라 음성데이터의 생성 시간과 순서번호를 이용하여 패킷의 순서와 미디어 간의 동기화를 지원해 준다. 그러나 RTP 프로토콜 헤더 정보의 타임스탬프, 순서번호, 그리고 수신측, 송신측의 통화품질 테이블을 통하여 전송시 네트워크 자원의 상태인 지연, 지터 그리고 패킷손실을 알아낼 수 있다. 특히 전송 도중 패킷손실 발생시 수신측에서 순서번호를 통하여 패킷손실률과 연속적인 손실패킷 수를 계산하여 송신측으로 피드백 정보를 RTCP 프로토콜을 통해 비주기적으로 전송하며, 피드백 정보를 수신한 송신측은 패킷손실 형태에 따라 부가정보를 전송하게 된다.

인터넷 트래피가 많이 발생시에 약 20%가 넘는 패킷손실률과 4개 이상의 연속패킷손실수가 나타나며 (그림 11, 12), 정적인 손실복구 방식의 경우, 1-type 부가정보만을 전송시 4개의 손실패킷 중 1개의 패킷만 복구 가능하므로, 한 패킷의 길이가 30 ms이면 90 ms의 음성정보가 손실되어 한음절 정도의 의미전달이 불가능해진다. 반면에 제안한 동적 손실복구 방식은 연속 패킷손실수가 4개인 경우 2-type 부가정보를 가지고 63% 정도의 손실패킷

을 복구하여 하나의 패킷 정도만 손실되므로 음질향상 및 의미전달의 문제를 해결할 수 있다. 또한 불연속적인 패킷손실이 많이 발생시는 1개 정도의 패킷은 부가정보의 복원을 통하여 42% 정도 손실패킷 복원이 가능하므로 음성정보의 끊김 현상을 방지할 수 있다.

동적 제어 알고리즘의 패킷손실 보상을 위해 RTCP 프로토콜을 통한 피드백 정보 전송과 송신측의 부가정보 전송 여부 등의 동적 부가 전송 방식은 재전송 방식에서 갖는 완전한 신뢰성 전송을 보장할 수 없으나, 실시간 전송의 중요한 품질 장애를 일으키는 많은 대역폭의 낭비 없이 손실된 패킷을 복구하는 장점 때문에 실시간 오디오 응용에 적합하다. 기존의 정적인 부가 전송은 동적으로 변하는 패킷 손실 특성과 상관없이 항상 동일한 부가적인 정보를 주 정보와 함께 전송을 하여 많은 대역폭을 낭비하여 결과적으로 네트워크에 많은 부하를 초래하며, 또한 단말기 상에서 주 정보와 부가정보를 생성, 분리 및 재생하는 프로세싱 타임을 초래하여 실시간 정보 송수신의 품질을 저하시켰다. 따라서 RTCP의 피드백 정보를 이용하여 동적인 부가 전송 방식을 적용함으로써 부가 데이터에 따른 대역폭 증가를 최소화하였다.

본 논문에서 제안한 동적 제어 알고리즘의 패킷 손실 보상을 위한 1-type, 2-type 부가정보 전송 방식은 수신 단말기 상에 프로세싱 타임을 증가시키는 경향이 발생하였으며, 추가적으로 수신 단말기 수신 버퍼의 동적인 제어방식을 통한 프로세싱 타임의 증가 및 수신 패킷의 음성 재생의 시간 향상 기술이 요구된다.

참고 문헌

1. Vocaltec Communication, <http://www.vocaltec.com>
2. D. S. Ko and J. S. Park, "Implementation of the real-time speech communication over the Korean Internet and quality assessment tests," *Proceedings of ICSP 97*, 661-666, Seoul, Korea, 1997.
3. ITU-T Rec. H.323, *Packet-Based Multimedia Communication System*, Feb. 1998.
4. AVT Working Group, <http://www-mice.cs.ucl.ac.uk/multimedia/misc/avt>
5. O. Hersent, *IP Telephony : Packet-Based Multimedia Communication System*, Addison Wesley, 316-326, 2000.
6. T. J. Koslas, T. J. Koslas, M. S. Borella, I. Sidhu, G. M. Shustor, J. Grabiec, and J. Mahter of 3Con, paper, "Real-time Voice over packet-switched network," *IEEE Network*, 12 (1), 18-27, Jan. 1998.
7. J. Rinde, "Telephony in the year 2005," *Computer Networks*, 31, 157-168, Feb. 1999.

8. P. Granstorm, PSTN/IP gateways and switches workshop, in Voice on the Net '99, Ericsson, 121-126, Apr. 1999.
9. W. Stevens, *TCP Illustrated*, I, Addison Wesley, 234-242, 1995.
10. J. Davidson, "Voice over IP network design," *CCIE '98*, Cisco system, 34-39, 1988.

저자 약력

• 김 무 중 (Mu Jung Kim)



1999년 2월: 대전대학교 정보통신공학과 (공학사)
2001년 8월: 대전대학교 정보통신공학과 (공학석사)
2001년 7월~현재: (주)언어과학 음성공학연구소 주
임연구원
※ 주관심분야: VoIP, TTS, 음성인식

• 권 철 홍 (Chul Hong Kwon)



1987년 2월: 서울대학교 전자공학과 (공학사)
1989년 2월: 한국과학기술원 전기·전자공학과 (공학
석사)
1994년 8월: 한국과학기술원 전기·전자공학과 (공학
박사)
1989년 1월~1997년 2월: 디지털 정보통신연구소
선임연구원
1997년 3월~현재: 대전대학교 컴퓨터·정보통신공
학부 (조교수)
1999년 8월~2000년 8월: 벨 연구소, 미국 루스트 (MTS)
※ 주관심분야: VoIP, TTS, 음성코딩, DSP