

論文 2003-40SP-6-8

G.729A와 SMV 음성부호화기를 위한 파라미터 직접 변환 방식의 상호부호화 알고리즘

(Transcoding Algorithm for SMV and G.729A Vocoders via Direct Parameter Transformation)

張 達 源 * , 徐 晟 豪 * , 李 善 一 * , 柳 昌 東 *

(Dalwon Jang, Seongho Seo, Sunil Lee, and Chang D. Yoo)

요 약

본 논문에서는 G.729A와 SMV 음성부호화기를 위한 새로운 파라미터 직접 변환 방식의 상호부호화 알고리즘을 제안한다. 상호부호화를 위하여 부가적인 복호화, 부호화 과정을 거쳐야하는 기존의 Tandem 방식과 달리 제안된 파라미터 직접 변환 방식에서는 양 음성부호화기가 음성을 부호화하기 위하여 공통적으로 사용하는 파라미터들이 직접 변환된다. SMV에서 G.729A로의 트랜스코딩에서는 LSP 변환, 피치 지연 변환, 낮은 전송률에서의 트랜스코딩 등의 알고리즘을 제안하고, G.729A에서 SMV로의 트랜스코딩에서는 LSP 변환, 피치 지연 변환, 전송률 결정 등의 알고리즘을 제안한다. 제안된 알고리즘을 다양한 방법으로 평가해본 결과 기존의 Tandem 방식과 비교하여 계산량과 지연 시간을 줄이면서도 동등한 음질 또는 향상된 음질을 구현함을 확인할 수 있었다.

Abstract

In this paper, a novel transcoding algorithm for the G.729A and the Selectable Mode Vocoder (SMV) vocoders via direct parameter transformation is proposed. In contrast to the conventional tandem transcoding algorithm, the proposed algorithm converts the parameters of one coder to the other without going through the decoding and encoding processes. In transcoder from SMV to G.729A, LSP conversion algorithm, pitch delay conversion algorithm and transcoding algorithm in lower rate are proposed, and in transcoder from G.729A to SMV, LSP conversion algorithm, pitch delay conversion algorithm and rate selection algorithm are proposed. Evaluation results show that while exhibiting better computational and delay characteristics, the proposed algorithm produces equivalent or improved speech quality to that produced by the tandem transcoding algorithm.

Keyword : transcoding, G.729A, SMV, transcoding, direct parameter transformation

I. 서 론

현재는 다양한 통신망들이 존재하고 또 새로이 생겨나고 있다. 각 통신망은 고유의 특성에 적합한 음성부호화 표준을 채택하여 사용하고 있다. 예를 들어 일반 유선 전화망은 펄스부호 변조(Pulse Code Modulation, PCM) 방식의 ITU G.711^[1] 음성부호화기를 사용하는 데에 반해 최근 활발히 보급되고 있는 Voice over

* 正會員, 韓國科學技術院 電子電算學科

(Dept. of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology)

接受日字:2003年6月18日, 수정완료일:2003年11月22日

Internet Protocol (VoIP)망에서는 Code Excited Linear Prediction (CELP)^[1]방식의 ITU G.723.1^[2]과 ITU G.729A^[3] 음성부호화기가 널리 쓰이고 있다. 또한 이동 통신 환경에서는 역시 CELP 방식인 Adaptive Multi Rate (AMR)^[4]와 Enhanced Variable Rate Codec (EVRC)^[5] 음성부호화기 등이 표준으로 사용되고 있다.

최근 유/무선 음성 통신망과 VoIP 방식을 이용하는 데이터 통신망간의 통합과 효율적인 연동이 중요한 과제로 제기되고 있다. 그러나 각각의 통신망은 서로 다른 음성부호화 표준을 사용하기에, 이것은 서로 호환이 되지 않는다. 그렇기 때문에 한 음성부호화기로 부호화된 신호를 다른 음성부호화기의 신호로 바꾸는 상호부호화(Transcoding) 과정이 필요하다. 상호부호화의 가장 기본적인 방법은 한 음성부호화기의 복호화기와 다른 음성부호화기의 부호화기를 연결하는 Tandem 방식이다. <그림 1(a)>는 Tandem 방식을 이용하여 음성부호화기 A와 음성부호화기 B간의 상호부호화를 수행하는 과정을 나타낸 것이다. 그림을 보면 알 수 있듯이 Tandem 방식에서는 먼저 원천 음성부호화기(Source Coder)로부터 전송된 비트열(Bit-stream)을 복호화(Decoding)하여 PCM 형식의 음성으로 합성한 후, 이를 다시 목적 음성부호화기(Target Coder) 형식으로 부호화(Encoding)하는 과정으로 상호부호화를 수행한다. 하지만 Tandem 방식으로 상호부호화를 수행할 경우, 불필요한 복호화와 부호화 과정 때문에 계산량과 지연 시간이 늘어나고, 음질 또한 저하되는 문제점이 발생한다. 이를 해결하고자 하는 것이 바로 파라미터

직접 변환 방식의 상호부호화 알고리즘이다. 이 알고리즘은 <그림 1(b)>와 같이 원천 음성부호화기의 복호화 과정과 목적 음성부호화기의 부호화 과정이 하나로 합쳐진 것이다

파라미터 직접 변환 방식의 상호부호화 알고리즘은 대상이 되는 음성부호화기들이 CELP 등의 같은 음성부호화 알고리즘에 기반하고 있어 음성을 부호화할 때에 같은 파라미터(Parameter) 집합을 사용할 때에 적용이 가능하다. 이 알고리즘은 음성을 완전히합성하지 않고 원천 부호화기의 복호화 과정의 일부만 사용하여, 파라미터를 추출해서 이것을 목적 부호화기의 부호화 과정에 이용한다. 이러한 직접적인 변환을 통해 Tandem 방식의 문제점인 계산량과 지연 시간 증가를 해결하고, 나아가 음질 저하 또한 막는 것이 파라미터 직접 변환 방식의 목표라 할 수 있다. 기존의 파라미터 직접 변환 방식에 기반한 상호부호화를 다룬 논문들은 [14-17]등이 있으며, Qualcomm사가 부분적으로 파라미터 직접 변환 방식을 이용하여 출원한 미국 특허도^[18] 있다.

본 논문은 G.729A 음성부호화기^[3]와 SMV 음성부호화기^[7-8]를 위한 새로운 파라미터 직접 변환 방식의 상호부호화 알고리즘을 제안한다. 두 음성부호화기는 CELP 기반 알고리즘을 사용한다. G.729A는 Conjugate Structure Algebraic Code-Excited Linear Prediction (CS-ACELP) 알고리즘을 기반으로 하고 있고, SMV는 extended Code Excited Linear Prediction(eX-CELP)^[9] 알고리즘을 기반으로 하고 있다. 두 음성부호화기로 부호화된 신호는 Line Spectral Pair(LSP)^[13] 또는 Line Spectral Frequency (LSF), 피치 지연(pitch delay)과 적응 코드북 이득(Adaptive Codebook Gain), 고정 코드북 벡터(Fixed Codebook Vector)와 고정 코드북 이득(Fixed Codebook Gain) 등의 정보를 가지고 있다. 이런 정보들을 가지고 목적 부호화기의 부호화 신호를 만들어내는 것이다.

본 논문은 다음과 같이 구성되어 있다. 먼저 II 장에서는 상호부호화의 대상이 되는 두 음성부호화기에 대하여 간략히 살펴본다. 그리고 III장과 IV장에서는 SMV에서 G.729A로의, 그리고 G.729A에서 SMV로의 상호부호화 알고리즘을 설명한다. V장에서는 양방향 상호부호화에 대해서 제안된 파라미터 직접 변환 방식의 상호부호화 알고리즘을 다양한 방법으로 평가한 결과를 제시하며, 마지막으로 VI장에서는 이 논문에 대한

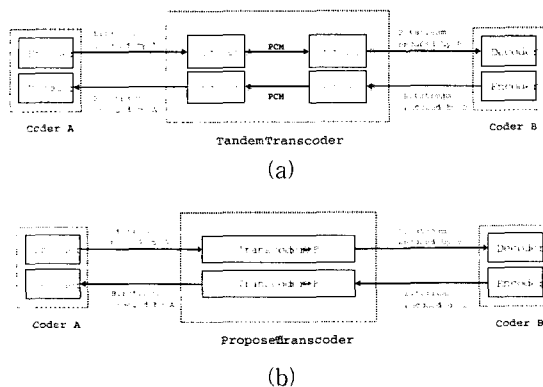


그림 1. 상호부호화 방식 (a) Tandem (b) 파라미터 직접 변환

Fig. 1. Transcoding Method. (a) Tandem (b) Direct Parameter Transformation

결론을 맺는다.

II. G.729A와 SMV 음성부호화기

1. G.729A 음성부호화기

G.729A는 Conjugate Structure Algebraic Code Excited Linear Prediction(CS-ACELP)를 기반으로 하는 ITU-T의 표준 음성부호화기이다. G.729A 음성부호화기는 G.729 음성부호화기⁶⁾를 계산량을 감소시키는 알고리즘을 적용하여 개선한 것이다. 부호화된 비트열은 G.729와 상호호환 가능하고, 내부 알고리즘만 다르다. G.729에서 계산량이 감소된 부분은, 지각 가중 필터, 개회로 피치 추정, 적응 코드북, 고정 코드북 검색, 그리고 후처리 필터이다.

G.729A 음성 부호화기는 음성신호를 프레임 단위로 합성에 의한 분석(Analysis by Synthesis)⁽¹⁰⁾을 이용한 선형예측 방법으로 부호화한다. 여기 신호는 ACELP 방법으로 양자화하며 여기 신호의 이득을 conjugate 구조의 벡터 양자화로 모델링한다. G.729A의 프레임 길이는 10ms이며 한 개의 프레임은 5ms 길이의 두 개의 부프레임을 가지고 있다. G.729A 음성 부호화기의 입력은 8kHz로 샘플링되어 16bit로 양자화된 PCM 신호이다. 입력신호는 전처리과정을 통해서 크기가 반으로 줄어들고 DC 성분이 제거된다. 전처리 과정을 거친 입력음성은 선형 예측 분석을 거치게 되고, 각 프레임마다 10차의 선형 예측 계수(Linear-Prediction Coefficient, LPC)가 구해진다. 선형 예측 분석 과정에서는 30ms 비대칭 윈도우를 입력 음성에 씌우는데, 이 때, 5ms의 예견 구간을 필요로 한다. 그리고, Levinson-Durbin 알고리즘을 이용해서 선형 예측 계수를 구한다. 선형 예측 계수는 양자화 에러 및 전송 과정에서의 에러를 줄이기 위해서 LSP의 형태로 변환되어 전송된다. 선형 예측 분석 후에는 여기 신호를 구하게 된다. 여기 신호는 적응 코드북 벡터(피치 지연)와 고정 코드북 벡터에 각 코드북의 이득을 곱하여 더한 형태로 모델링된다. 피치 지연은 개회로 피치 검색과 폐회로 피치 검색 과정을 통해서 각 부프레임마다 하나씩 구해지며 1/3 단위의 정밀도를 가진다. 개회로 검색에서 피치 지연의 범위를 정하고, 폐회로 피치 검색 과정에서 정확한 피치 지연을 찾는다. 고정 코드북은 Interleaved Single-Pulse Permutation(ISPP) 구조에 기반하고 각

부프레임당 4개의 펄스를 가진다.

2. SMV 음성부호화기

SMV는 extended Code Excited Linear Prediction (eX-CELP)을 기반으로 만들어졌다. SMV는 8.55, 4.0, 2.0, 0.8kbps, 총 네 개의 전송률을 가지고 있으며, 프레임 단위로 전송률을 결정해서 부호화한다. 위의 네 개의 전송률을 각각 Rate 1(full-rate), Rate 1/2(half-rate), Rate 1/4(quarter-rate), Rate 1/8(eighth-rate)라고 한다. 그리고 8.55kbps와 4.0kbps의 전송률의 경우 type 0와 type 1, 두 가지의 type을 가지고 있다. 전송률과 type을 결정하는데 중요한 요소로 프레임 클래스(frame class)가 있다. SMV는 각 프레임을 입력 음성의 특성에 따라서, 묵음(silence), 잡음(noise-like), 무성음(unvoiced), 변화(onset), 일정하지 않은 유성음(non-stationary voiced), 일정한 유성음(stationary voiced), 총 6가지의 프레임 클래스로 분류한다. 프레임 클래스에 따라서 결정되는 전송률이 달라지고, 일정한 유성음으로 분류되었을 경우, type 1로 결정되어서 type 0과는 다른 부호화 과정을 거치게 된다. 이 경우, 고정 코드북에 대한 정보에 많은 비트가 배당된다.

Rate 1과 Rate 1/2, 두가지 전송률에서는 CELP의 기본 파라미터 5개와 type에 대한 정보를 가진다. 반면에, Rate 1/4과 Rate 1/8에서는 랜덤하게 생성된 신호를 여기 신호로 사용하기 때문에, 많은 정보가 전송되지 않는다. Rate 1/4에서는 LSF와 여기 신호의 이득, 그리고 여기 신호에 가해지는 필터링에 대한 정보만을 가진다. Rate 1/8에서는 LSF와 음성의 에너지 정보만을 가진다.

SMV는 4개의 동작모드를 가진다. 이것은 내부의 알고리즘에서 결정하는 것이 아니라 외부의 통신 환경에 의해서 결정되는 것이다. 이 모드에 따라서 SMV 음성 부호화기는 전송률을 결정하게 된다. 네 개의 모드는 Mode 0(premium mode), Mode 1(standard mode), Mode 2(economy mode), Mode 3(capacity-saving mode)이다. 모드의 이름에서 알 수 있듯이 모드의 번호가 줄어들수록 높은 전송률을 많이 사용해서 좋은 품질의 음성을 전송한다. 음질이 좋아지는 대신에 높은 평균 전송률을 가지게 된다.

SMV에서 사용하는 프레임의 길이는 20ms이고 부프레임의 개수와 그 길이는 프레임의 전송률에 따라서 달라지게 된다. Rate 1에서는 4개의 부프레임을 사용하

고, Rate 1/2에서는 type에 따라서 2개 또는 3개의 부프레임을 사용한다. 그리고 Rate 1/4에서는 10개의 부프레임을 사용하고, Rate 1/8에서는 부프레임이 없다. 선형 예측 분석 과정에서는 10ms의 예견 구간을 가지는 윈도우를 사용하고 10차의 선형 예측 계수를 구해서 LSF로 변환해서 양자화한다.

III. SMV에서 G.729A 음성부호화기로의 상호부호화

1. 전체적인 구조

<그림 2>는 제안된 파라미터 직접 변환 방식의 상호부호화 알고리즘의 구조도를 나타낸 것이다. 이것은 SMV 신호 중에서 높은 전송률(Rate 1, Rate 1/2 Rate 1/4)에서 생성된 비트열에 대한 상호부호화이다. 앞 장에서 설명하였듯이 SMV는 4개의 전송률을 가지는데, SMV의 낮은 전송률(Rate 1/8)에 대해서는 간단히 상호부호화 가능하기 때문에, <그림 2>와는 다른 방식으로 상호부호화를 한다. 이에 대해서는 뒤에서 따로 언급할 것이며, 특별한 언급이 없을 경우에는 SMV의 높은 전송률로 부호화된 비트열 신호의 상호부호화에 대한 설명이다. 전송률에 따라서 다르기는 하지만, SMV는 CELP의 구조를 따르고 있기 때문에 SMV의 신호를 파라미터로 변환하면 LSP에 관련된 값들과 여기 신호에 관련된 값들이 얻어진다. LSP는 직접적으로 변환 가능하고 여기 신호는 SMV의 파라미터들을 복호화하

SMV Bit stream

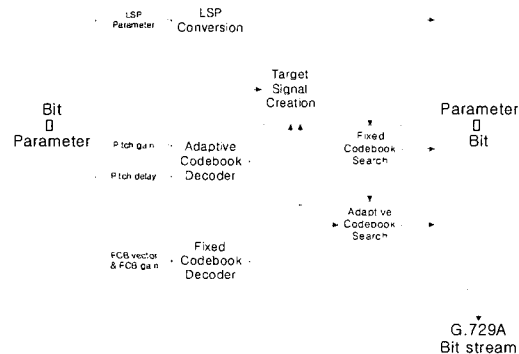


그림 2. SMV → G.729A 상호부호화의 블록도
Fig. 2. Block diagram of SMV → G.729A transcoding.

여 생성한 음성신호 - 후처리 과정을 거치지 않기 때문에 거친 음성이고 이 블록 다이어그램에서는 target signal로 표시를 분석해서 찾는 과정을 거친다. 물론 이 분석 과정에서 SMV의 파라미터들을 이용해서 효율성을 높인다. 이렇게 얻어진 G.729A의 파라미터는 비트열로 변환 후 전송된다.

2. LSP 변환

SMV와 G.729A에서 선형 예측 계수는 똑같이 10차의 계수를 가지고, Levinson-Durbin 알고리즘을 사용하는 선형 예측 분석 방법은 같다. 양 음성부호화기 모두 전처리 과정을 거친 신호가 선형 예측 분석을 거치게 되는데, 이 때, 가장 먼저 가중 윈도우를 신호에 씌운다. 두 음성부호화기의 가중 윈도우는 서로 다르지만, 가중 윈도우를 거친 신호가 같다면, SMV에서 구해진 LSP 파라미터는 G.729A에서 그대로 사용 가능하다. 즉, 선형 예측 분석을 할 때 입력에 어떻게 가중 윈도우가 사용되는지를 살펴보고, 그것을 보완하는 과정을 거치면 LSP의 상호부호화를 할 수 있다.

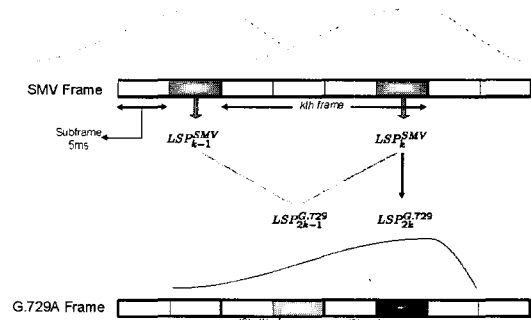


그림 3. SMV에서 G.729A로의 LSP 변환
Fig. 3. LSP transcoding from SMV to G.729A.

<그림 3>은 SMV의 k번째 프레임과 그 전 프레임에 대해서 선형예측 분석 과정에서 사용한 가중 윈도우를 그리고, SMV의 k번째 프레임에 대응하는 G.729A의 두 프레임에 대해서 윈도우를 그린 것이다. SMV는 20ms, G.729A는 10ms의 프레임 길이를 가지기 때문에 SMV의 한 프레임이 G.729A의 두 프레임으로 변환된다. SMV에서는 세 번의 선형 예측 분석을 하지만, 여기서는 그 중 LSP 형태로 전송되는 선형 예측 계수를 찾을 때, 즉 상호부호화에 사용되는 LSP 파라미터를 구할 때 사용하는 윈도우만을 그린 것이다. 그리고 G.729A에 대해서는 각각의 프레임에 대응하는 윈도우를 그렸다.

그리고 각 프레임은 짧은 선으로 나누어지고, 부프레임은 얇은 선으로 구분된다. 부프레임 중에서 색칠이 된 부프레임은 선형 예측을 위한 윈도우의 중심을 나타낸다.

SMV의 k 번째 프레임에서 사용하는 윈도우와 G.729A에서 $2k$ 번째 프레임에서 사용하는 윈도우는 시작과 끝의 위치는 다르지만, 그 모양이 상당히 비슷하고, 같은 위치를 중심으로 한다. 그렇기 때문에 SMV에서 G.729A로 변환하는 상호부호화기에서는, SMV의 k 번째 프레임의 LSP 값을 G.729A의 $2k$ 번째 프레임의 LSP 값으로 그대로 사용한다. 그리고 G.729A의 $(2k-1)$ 번째 프레임은 SMV의 $(k-1)$ 번째 프레임과 k 번째 프레임의 중간에 위치하기 때문에, 두 SMV 프레임의 LSP 값들의 평균을 G.729A의 $(2k-1)$ 번째 프레임의 LSP 계수로 하였다. 이 두 가지 LSP 변환을 식으로 표현하면 다음과 같다.

$$LSP_{2k-1}^{G.729A} = \frac{1}{2}(LSP_{k-1}^{SMV} + LSP_k^{SMV}) \quad (1)$$

$$LSP_{2k}^{G.729A} = LSP_k^{SMV} \quad (2)$$

그리고, LSP 변환 전에 먼저 해 주어야 하는 것은 SMV의 LSF를 LSP로 변환하는 과정이다. G.729A는 LSP를 저장하지만, SMV는 LSF를 저장한다. 하지만, LSF는 간단하게 LSP로 변환할 수 있고, G.729A의 LSP 계수를 만들기 위해서 SMV의 LSF를 LSP로 변환하여 사용한다.

3. 피치 지연 변환

<그림 4>는 두 음성부호화기의 피치 지연의 경향을 알아보기 위한 것으로 같은 음성에 대해서 SMV 음성부호화기에서 계산된 피치 지연과 G.729A 음성부호화기에서 계산된 피치 지연을 각 프레임마다 비교해서 그린 것이다. 세로축은 상대적인 크기이며, 가로축은 프레임이다. 피치 지연의 부샘플(sub-sample) 단위는 무시하고 정수 단위만을 나타냈다. 그리고, 피치 지연은 각 부프레임마다 구해지나, 여기서는 한 프레임에 대해서 피치 지연들의 평균을 계산하여 그것을 그 프레임의 피치 지연으로 정해서 그렸다. SMV에서 피치 지연 검색을 하지 않는 영역(Rate 1/4나 Rate 1/8로 결정된 프레임)에서는 피치 지연을 0으로 두었다.

SMV의 피치 지연이 0인 영역을 제외하고는 두 음성부호화기의 피치 지연이 많은 부분 일치함을 <그림 4>

에서 확인할 수 있다. 그래서 SMV 비트열에서 나온 피치 지연을 그대로 사용해 보았으나 결과음성의 음질이 상당히 떨어지는 것을 확인할 수 있었다. 따라서, SMV의 피치 지연을 그대로 사용할 수 없다. 하지만, 두 음성부호화기에서 나오는 피치 지연이 상당히 비슷하다는 점을 최대한 이용하여 SMV의 피치 지연 부근에서 피치 지연의 폐회로 검색을 하여 피치 지연을 구하는 알고리즘을 사용하였다. 즉 SMV의 피치 지연이 G.729A에서 피치 지연을 찾기 위해서 수행하는 개회로 검색의 결과를 대신한다고 생각할 수 있다. 이 방법으로 개회로 피치 검색 과정을 제거하여 개회로 피치 검색 과정만큼의 계산량의 감소를 가져올 수 있으며, 이에 따른 음질의 저하는 극히 적다.

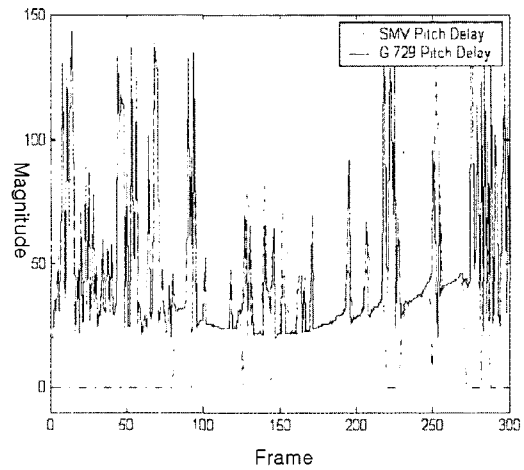


그림 4. G.729A의 피치 지연(실선)과 SMV의 피치 지연(점선)의 비교

Fig. 4. Comparison of the pitch delay of G.729A (solid line) and the pitch delay of SMV(dotted line).

4. 이득 변환

<그림 5>는 같은 음성에 대해서 SMV 음성부호화기에서 계산된 고정 코드북 이득과 G.729A 음성부호화기에서 계산된 고정 코드북 이득을 비교하여 그린 것이다. <그림 4>와 마찬가지로 여기에서도 한 프레임 당 하나의 평균값을 구해서 나타내었고, 역시 가로축은 프레임이며, 세로축은 상대적인 크기이다. 그리고 SMV에서 Rate 1/8 구간에 대해서는 gain을 0으로 하였다.

<그림 5>에서 확인할 수 있듯이 두 값은 상당히 유사하다. SMV의 고정 코드북 이득을 적절한 선형 보간

과 크기 보정 과정을 거치면, G.729A의 신호와 아주 흡사한 신호를 만들 수 있다. 이렇게 두 음성부호화기의 고정 코드북 이득은 매우 비슷하기 때문에 간단한 변환을 통해서 상호부호화가 가능하다.

고정 코드북 이득과는 달리, 적응 코드북의 이득은 두 부호화기에서 구한 것에 많은 차이가 있다는 것을 <그림 6>에서 확인할 수 있다. <그림 6> 역시 같은 음성에 대해서 SMV 음성부호화기에서 계산된 적응 코드북의 이득과 G.729A 음성부호화기에서 계산된 적응 코드북의 이득을 비교해서 그린 것이다. <그림 4>의 경우와 같이 <그림 6>에서도 한 프레임 당 하나의 평균값을 구해서 그렸고 가로축은 프레임의 수이며, 세로축은 상대적인 크기이다. 그리고 SMV에서 적응 코드북 이득을 구하지 않는 구간에 대해서는 이득을 0으로 하였다.

이렇게 두 음성부호화기에서 나온 적응 코드북의 이득이 많은 차이가 나고 연관성을 찾기 힘들다. 따라서 SMV의 비트열에서 나온 적응 코드북 이득을 G.729A로 상호부호화하는 데 사용하는 것은 무리가 있다.

그런 문제로 적응 코드북의 이득은 간단한 방법으로 상호부호화할 수 없고, 새롭게 구해야 한다. 그런데, G.729A 부호화기에서는 고정 코드북 이득과 적응 코드북 이득을 동시에 계산한다. 그래서 이 경우 SMV의 비트열에서 얻은 고정 코드북 이득은 활용 가능하지만, 적응 코드북 이득을 새로 구해야 하기 때문에 고정 코드북 이득의 활용 가능성이 적다. 즉, 고정 코드북 이득을 이용한 상호부호화에서 얻을 수 있는 계산량 저하

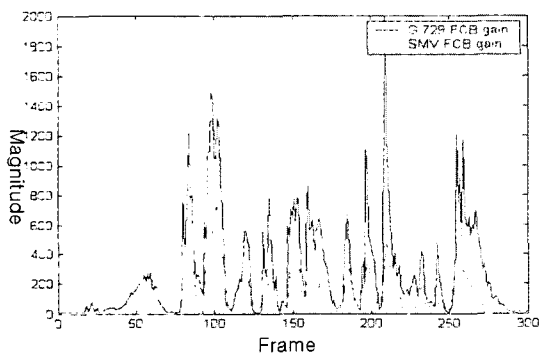


그림 5. G.729A의 고정 코드북 이득(실선)과 SMV의 고정 코드북 이득(점선)의 비교
Fig. 5. Comparison of the FCB gain calculated by G.729A (solid line) and FCB gain calculated by SMV (dotted line).

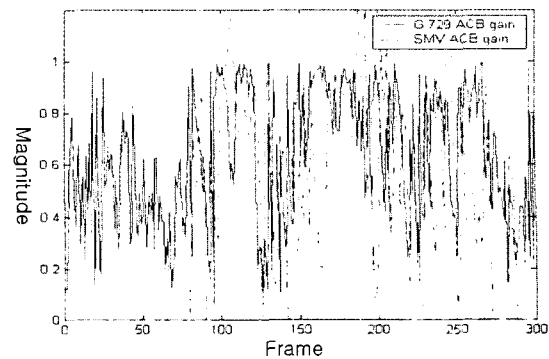


그림 6. G.729A의 적응 코드북 이득(실선)과 SMV의 적응 코드북 이득(점선)의 비교
Fig. 6. Comparison of the ACB gain calculated by G.729A (solid line) and ACB gain calculated by SMV (dotted line).

의 정도가 미미하다. SMV에서 얻은 고정 코드북 이득도 버리고, 두 이득을 다 새롭게 계산하였다.

5. Rate 1/8에서의 상호부호화

지금까지 설명한 모든 상호부호화 과정들은 SMV의 높은 전송률에서 G.729A로의 상호부호화 과정에 관한 것이다. 낮은 전송률(Rate 1/8)로 부호화된 비트 신호는 위에서 설명한 모든 과정을 거칠 필요도 없고, 모두 거친다는 것은 불필요한 계산량의 증가를 낳는다. Rate 1/4과 Rate 1/8의 경우, 부호화기에서 부호화되어 비트로 전송될 때부터 신호가 갖고 있는 정보가 한정되어 있기 때문이다. 낮은 전송률의 신호에 대해서는 다음과 같은 과정을 통해서 상호부호화가 이루어진다.

SMV에서 Rate 1/8로 선택된 프레임은 프레임 분류 과정에서 묵음이라고 결정된 프레임이다. Rate 1/8로 전송된 프레임에서 비트로 전송되는 정보는 LSP 계수와 프레임의 에너지 정보 뿐이다.

Rate 1/8의 경우, LSP를 상호부호화하는 과정과 이득값을 상호부호화하는 과정만 거치도록 한다. LSP를 상호부호화하는 것은 앞에서 설명한 과정과 동일하게 이루어진다. 에너지는 적절한 변환 과정을 거쳐서 G.729A의 이득을 만들어내야 한다. 그러나 이 경우 Rate 1/8인 프레임은 묵음 구간이라는 성질을 이용해서 이득을 간단히 결정할 수 있다. G.729A 부호화기에서는 고정 코드북 이득과 적응 코드북 이득을 7bit로 벡터 양자화를 한다. 이득이 가질 수 있는 값은 이미 결정되어 있는 128가지 값 중 하나이다. 이 128가지 값 중 묵음구간에서 선택되는 특정한 이득이 존재한다. 이

사실은 같은 음성에 대해서 각각 SMV와 G.729A로 부호화했을 때, SMV에서 묵음구간이라고 판단되는 구간에서의 G.729A의 이득을 확인함으로써 알 수 있다. 그런 특정 이득이 존재하기에, Rate 1/8인 신호에 대해서는 G.729A로 상호부호화할 때, 묵음구간에서 선택되는 특정 이득을 그대로 사용하면 된다. 이 과정에서 음질의 저하가 있긴 하지만, 그 영향은 크지 않다. 이 과정은 <그림 7>에 나타나 있다. Rate 1/8인 프레임에서도 똑같이 타겟 신호를 만들지만, 이것은 다음 프레임을 위한 것이고 현재 프레임의 부호화 과정에서는 사용되지 않는다. 피치 지연과 고정 코드북은 정하지 않고, 이전 프레임의 값들이 그대로 사용된다.

Rate 1/4로 결정된 프레임에 대해서도 이와 같은 간단한 상호부호화가 가능할 수 있다. Rate 1/4으로 부호화되어 전송되는 프레임은 묵음구간은 아니지만, 잡음 신호이거나, 무성음 구간, 그리고 무성음에서 유성음으로 변화하는 구간이다. 여기서 전송되는 정보는 LSP 계수들과 무성음에 해당하는 여기 신호의 대략적인 형태이다. 적응 코드북에 관한 것은 전송되지 않고, 여기 신호는 고정 코드북 벡터의 형태로 정보가 가진 않지만, 10개의 부프레임에 대한 이득 정보와 여기신호에 행해지는 필터링에 관한 정보 형태로 전송된다. 이것들을 G.729A의 파라미터 형태로 비꾸는데, 유성음에 대한 정보가 없는 신호이기 때문에 피치 지연과 적응 코드북 이득은 필요가 없다. 그래서 이것들을 구하는 과

정은 생략할 수 있다. 하지만, 이 경우 줄일 수 있는 계산량은 얼마 되지 않지만, 음질의 저하가 크다. 상호부호화기의 피치 검색은 SMV의 피치 지연으로 개회로 피치 지연 검색의 결과를 대체하였기에, 폐회로 검색만 수행하게 되는데, 폐회로 검색은 아주 작은 범위에서만 계산하므로, 계산량이 미미하다. 그래서 Rate 1/4로 결정된 프레임의 상호부호화에서 피치 지연 검색 과정 전체를 생략함으로써 계산량을 줄일 수는 있지만, 그렇게 하지 않고 일반적인 방법을 그대로 사용했다. 그리고 실험적으로 확인할 결과, Rate 1/4로 결정된 프레임에서는 Rate 1/8로 결정된 프레임과는 달리 하나의 이득으로 결정되는 것이 아니고, 다양한 이득을 가질 수 있다. 따라서 Rate 1/8로 결정된 프레임의 상호부호화와 같이 이득을 특정 값으로 고정시킴으로써 계산량을 줄이는 알고리즘은 사용할 수는 없다.

IV. G.729A에서 SMV 음성부호화기로의 상호부호화

1. 전체적인 구조

G.729A에서 SMV로의 상호부호화기의 전체적인 구조는 <그림 8>과 같다. SMV에서 G.729A로의 상호부호화 과정과 전체적인 구조가 상당히 비슷하다. LSP는 직접적인 변환을 통해서 얻을 수 있고 여기 신호는 새로 찾는 과정을 거친다. 하지만, 여기서는 한 가지 크게 다른 점이 있다. 그것은 G.729A에 비해서 SMV의 특징적인 과정이라고 할 수 있는 전송률과 type을 결정하는

SMV Bit stream

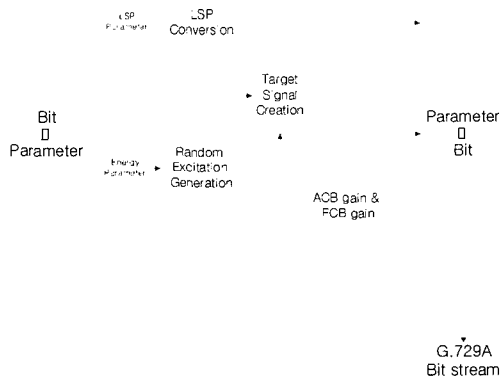


그림 7. SMV(Rate 1/8) → G.729A 상호부호화의 블락도
Fig. 7. Block diagram of SMV(Rate 1/8) → G.729A transcoding.

G.729A Bit stream

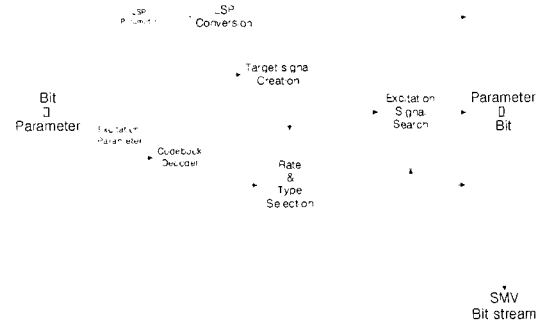


그림 8. G.729A → SMV 상호부호화의 블락도
Fig. 8. Block diagram of G.729A → SMV transcoding.

과정이다. SMV에서는 각 프레임마다 전송률과 type을 결정하고 그것들에 따라서 서로 다른 여기 신호를 생성하고, 양자화 과정도 다르다. 이 상호부호화기에서도 SMV 부호화 과정과 마찬가지로 이런 과정을 거쳐야 한다. 전송률과 type을 결정하는 과정은 입력으로 들어온 G.729A의 파라미터 중 여기 신호에 관한 것과 복호화해서 구해진 음성신호를 이용한다.

G.729A로 부호화된 신호는 전송률이 다르거나 다른 모드로 동작하는 경우가 없이 한 가지 종류의 신호만 있으므로 상호부호화 과정은 항상 <그림 8>과 같은 과정을 거친다.

2. 전송률과 type 결정 과정

상호부호화기에서 전송률과 type 결정 과정에 앞서 SMV에서는 이 과정이 어떻게 행해지는지 알아볼 필요가 있다. 전송률을 결정하기 위해서는 입력을 음성의 특성에 따라서 분류하는 프레임 분류 과정이 선행된다. 그리고 프레임 분류 과정의 결과를 이용해서 type도 결정된다. 프레임 분류 과정과 전송률 결정 과정을 수행하기 위해서는 여러 가지 계산 결과가 필요하다. <그림 9>에서 그려진대로 그런 결과들을 선형 예측 분석 과정, VAD 과정, 개회로 피치 분석 과정 등에서 얻어진다. <그림 9>와 같은 과정을 거쳐서 프레임 분류 과정과 전송률 결정 과정이 수행된다. <그림 9>에서 확인할 수 있듯이 SMV의 전송률 결정 과정은 선형 예측 분석 과정(LP analysis)과 개회로 피치 분석 과정(open-loop pitch detection)의 결과가 필요하다. 그러나, 이 과정들은 SMV에서 G.729A로의 상호부호화 알고리즘에서도 살펴보았듯이 상호부호화 알고리즘에서 생략 가능한 과정들이다. SMV의 전송률 결정 알고리즘을 그대로 사용할 경우, 이 과정들을 생략 불가능하기 때문에, 상호부호화의 효율이 떨어진다고 생각할 수 있다. 그리고 상호부호화기는 일반적인 음성부호화기와는 다르게, 이미 CELP 과정을 거쳐서 얻어진 파라미터들을 정보로 가지고 있고 이런 정보들을 이용하면 간단히 전송률과 type을 결정할 수 있고 그에 따라 계산량을 줄일 수 있다. 그래서 상호부호화에 적합한, 선형 예측 분석 과정과 개회로 피치 분석 과정이 없고 CELP 파라미터의 정보를 이용하는, 새로운 전송률 결정 과정이 필요하다.

상호부호화기에서 전송률 결정 과정은 CELP 파라미터와 복호화된 음성 신호를 입력으로 가진다. 이것들을

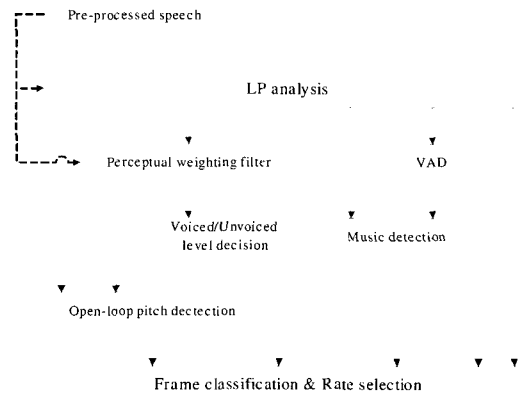


그림 9. SMV의 전송률 결정과정 블록도

Fig. 9. Block diagram of SMV rate selection algorithm.

이용해서 프레임 분류과정을 수행하고 그 결과를 이용해서 전송률과 type을 결정하는 것이 새로운 전송률 결정 과정의 핵심이다. <그림 8>에서 나오듯이, LSP 정보는 이용되지 않고, 여기 신호만 이용된다.

프레임 분류 과정은 <그림 10>과 같이 수행된다. SMV 부호화기에서는 6가지로 프레임을 분류하지만, 상호부호화기에서는 그 중 잡음을 제외하고 5가지 프레임 클래스로 분류해서 이것으로 전송률과 type을 결정한다.

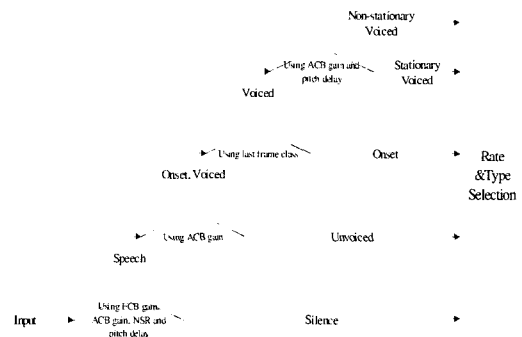


그림 10. 제안된 전송률 결정과정 알고리즘의 블록도

Fig. 10. Block diagram of proposed rate selection algorithm.

고정 코드북 이득(FCB gain)과 적응 코드북 이득(ACB gain), 잡음 대 신호비(NSR), 그리고 피치 지연(pitch delay)를 사용해서 음성(Speech)과 묵음(Silence)을 분류한다. 고정 코드북 이득과 적응 코드북 이득은 몇가지 선형 과정을 거친 다음 크기를 이용해서 분류

하고, 잡음 대 신호비는 일정 값 이상이 되면, 묵음으로 분류한다. 그리고 피치 지연이 편차가 큰 경향을 나타내면, 묵음으로 분류한다. 음성 구간에서는 적응 코드북 이득의 크기를 사용해서 무성음과 유성음(Voiced)으로 분류하고, 유성음 중에서 지난 프레임의 프레임 클래스가 무성음인 경우에만 변화(Onset)로 결정한다. 변화가 아닌 유성음은 일정한 유성음(Stationary voiced)과 일정하지 않은 유성음(Non-stationary voiced)으로 나눠 지는데, 프레임 내의 적응 코드북 이득값과 피치 지연의 변화에 따라서 나누어진다.

이런 모든 분류 과정은 CELP 파라미터를 몇 가지 선형적인 변형 후에 고정된 문턱값과의 비교를 통해서 이루어지는데, 문턱값들은 다양한 실험 음성을 대상으로 평균 전송률이 크게 달라지지 않는 범위 내에서 음질이 가장 좋아지도록 결정한다.

3. LSP 변환과 의도적인 지연

SMV의 한 프레임이 G.729A의 한 프레임보다 두 배의 길이를 가지므로 상호부호화기에서 한 개의 SMV 프레임을 만들 때 두 개의 LSP 계수들을 입력으로 받게 된다. <그림 11>은 SMV에서 G.729A로 상호부호화할 때와 마찬가지로 선형 예측 분석에 사용하는 가중 윈도우를 그린 것이다. <그림 11>에서 위는 G.729A의 프레임이고 아래는 SMV의 프레임이다. SMV의 k번째 프레임을 얻기 위해서 G.729A의 (2k-1), (2k)번째 프레임이 사용된다. 그리고, SMV 음성부호화기에서는 각각 다른 윈도우를 사용해서 3번의 선형 예측 분석을 하고 3가지의 선형 예측 계수 조합을 구하게 되는데, <그림 11>에 나타난 것은 3가지 중 가장 중요한, 실제로 양자화되어 비트열로 생성되는 선형 예측 계수들을 생성하는데 사용되는 윈도우이다. 다른 두 가지의 선형 예측 계수들은 G.729A의 LSP 계수들로 구해진 선형 예측 계수를 선형 보간해서 구한다.

선형 예측 분석에 사용하는 윈도우의 위치를 살펴보고 입력으로 받아들인 G.729A의 두 가지 LSP 계수들을 선형 결합하여 SMV의 LSP 계수들을 생성해 낸다. 여기서는 G.729A의 두 LSP 계수들의 평균으로 구할 수 있다. 이것은 윈도우에서 중심이 되는 위치를 이용하는 것이다. <그림 11>에서 G.729A의 윈도우는 G.729A 프레임에서 3번째, 5번째 부프레임이 중심이고, SMV의 윈도우는 4번째 부프레임이 중심이 된다. 그래서 G.729A의 두 LSP 계수를 1:1로 선형보간해서 SMV

의 LSP 계수들을 구할 수 있다. 아래의 식과 같이 적을 수 있다.

$$LSP_k^{SMV} = \frac{1}{2} (LSP_{2k-1}^{G.729A} + LSP_{2k}^{G.729A}) \quad (3)$$

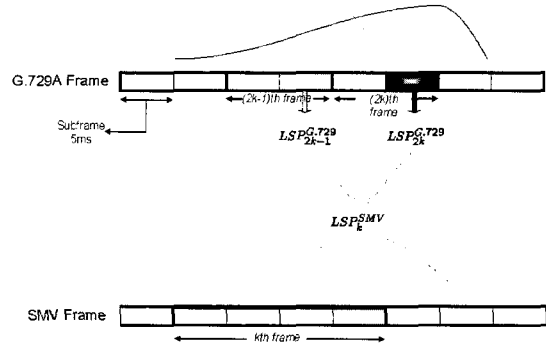


그림 11. G.729A에서 SMV의 LSP 변환
Fig. 11. LSP transcoding from G.729A to SMV.

<그림 11>을 보면, G.729A와 SMV의 프레임이 서로 맞지 않게 그려져 있다. G.729A의 (2k-1), (2k)번째 프레임이 SMV의 k번째 프레임보다 5ms만큼 뒤의 음성을 부호화한다. 이것은 상호부호화시 의도적으로 지연을 넣은 것이다. SMV에서 G.729A로의 부호화처럼 두 프레임의 위치가 서로 맞게 하면 지연 시간은 5ms 더 줄일 수 있지만, 음질이 나빠진다. 이것은 여기 신호를 찾는 과정에서 예견구간을 필요로 하기 때문이다. 지연 시간을 5ms만 줄이면서 음질을 좋게 하는 방향으로 알고리즘을 완성하였다.

4. 피치 지연 변환

SMV에서 G.729A로의 상호부호화 과정에서 살펴본 바와 같이 두 음성부호화기에서 나오는 피치 지연은 상당히 유사한 모습을 보인다. 하지만, SMV에서 G.729A로의 상호부호화에서와 마찬가지로 피치 지연을 그대로 사용할 경우, 음질의 저하가 생기게 되고, 그에 따라서 G.729A에서 SMV로의 상호부호화기에서도 앞의 경우와 마찬가지로 G.729A의 피치 지연의 정수값을 개회로 분석을 대신하는데 사용한다.

그런데, 이 경우 한 가지 문제점이 있다. G.729A 음성부호화기가 찾아내는 피치 지연은 약간의 문제를 가지고 있는데, 그것은 일정한 유성음의 경우, 실제 피치 지연의 두 배, 세 배가 되는 값들을 피치의 값들로 결정할 수 있다는 것이다. <그림 12>는 SMV의 개회로

검색에서 얻어진 피치 지연과 G.729A에서 최종적으로 결정한 피치 지연의 정수값을 비교해서 그린 것이다. 음성 구간 피치 지연의 변화가 적은 구간이 음성 구간이다 - 일 경우 두 피치 지연이 전체적으로 상당히 유사함을 확인할 수 있지만, 왼쪽에 동그라미로 표시한 부분을 자세히 보면, 차이를 발견할 수 있다. 이것은 피치 값의 두 배 또는 세 배가 G.729A의 피치 지연이 된 것이다. 이런 것을 그대로 사용하여 개회로 검색을 대신하게 하면 상호부호화기에서도 부정확한 피치 지연 값을 결정하게 되고, 이것은 심각하진 않지만 Tandem 한 상호부호화기에 비해서 음질의 저하를 초래한다.

이런 문제점을 제거하기 위해서 피치 지연을 이전 값들과 비교하는 방법을 사용한다. G.729A의 비트열에서 받아들인 피치 지연을 이전의 몇 개 값들과 계속 비교해서 이전 값들이 일정하고, 현재값이 이전 값들에 비해서 두 배나 세 배가 되었을 경우, 현재값이 실제 피치 지연의 두 배 또는 세 배가 된 값이라고 생각할 수 있다. 현재값을 앞의 값들과 같게 만들어줌으로써

이런 문제를 해결할 수 있었다. 앞의 세 개의 피치 지연값과 현재값의 반 또는 1/3과의 편차의 크기를 이용해 이것이 아주 작을 경우 피치를 반 또는 1/3으로 만들어주는 방법을 사용했다. 그 결과 <그림 12(하)>와 같이 피치 지연이 변화하였고, 이 피치 지연으로 개회로 검색을 대체했을 때, 이런 과정을 거치지 않은 <그림 12(하)>의 피치 지연으로 개회로 검색을 대체했을 때에 비해서 음질이 약간 향상되는 것을 확인할 수 있었다.

V. 성능 평가

본 장에서는 제안된 파라미터 직접 변환 방식의 상호부호화 알고리즘의 성능을 다양한 지표를 이용하여 평가한다. 성능 평가는 계산량, 지연 시간, 객관적 음질 평가로 이루어져있다. 성능 평가를 위하여 제안된 파라

표 1. 계산량 비교(SMV → G.729A)

Table 1. Comparison of computational complexity(SMV → G.729A).

입력	Mode	Tandem	Proposed	감소율
남	0	10.25	5.19	49.4%
	1	10.47	5.56	46.9%
	2	10.79	5.87	45.9%
	3	10.92	6.01	45.0%
여	0	10.32	5.34	48.3%
	1	10.71	5.90	44.9%
	2	10.95	6.13	44.0%
	3	10.99	6.18	43.8%

표 2. 계산량 비교(G.729A → SMV)

Table 2. Comparison of computational complexity(G.729A → SMV).

입력	Mode	Tandem	Proposed	감소율
남	0	27.69	23.19	16.3%
	1	25.90	20.26	21.8%
	2	25.55	20.02	21.6%
	3	25.51	20.00	21.6%
여	0	29.24	24.29	16.9%
	1	27.06	21.40	20.9%
	2	26.80	21.16	21.0%
	3	26.82	21.16	21.1%

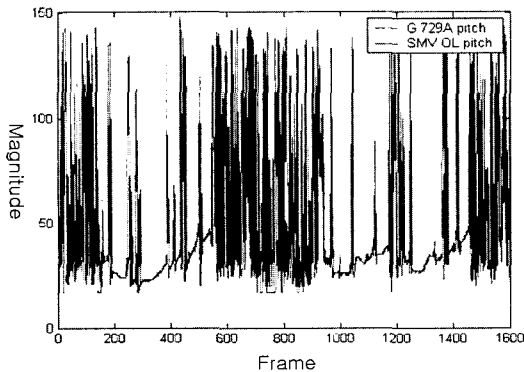
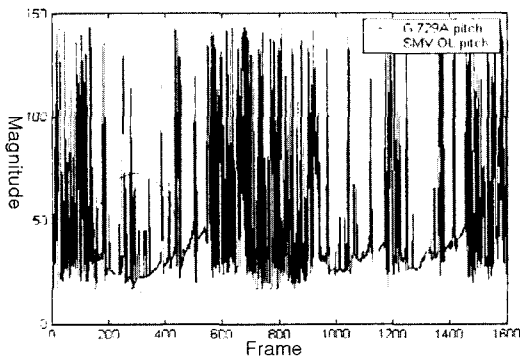


그림 12. G.729A의 피치 지연의 문제점(위)과 해결한 결과(아래)

Fig. 12. Problem of pitch delay of G.729A(top) and Result(bottom).

미터 직접 변환 방식의 상호부호화 알고리즘을 C 언어를 이용하여 고정 소수점 형태로 구현하였다. 그리고 입력 음성으로는 남자 4명, 여자 4명에 의해 발음된 8초 길이의 한국어 음성 16 가지가 모든 모의 실험에서 공통적으로 사용되었다.

SMV는 4가지의 모드를 가지기 때문에, 각 모드들에 대해서 성능 평가를 하였고, half rate max 모드일 경우는 성능 평가를 하지 않았다.

1. 계산량

제안된 알고리즘을 평가하기 위해서 Weighted Million Operations Per Second (WMOPS)를 측정하였다. <표 1, 2>는 제안된 파라미터 직접 변환 방식의 계산량과 Tandem 방식의 계산량을 비교한 것이다. <표 1, 2>를 통해서 제안된 파라미터 직접 변환 방식이 계산량을 많이 줄인다는 것을 확인할 수 있다. SMV에서 G.729A로의 상호부호화에서는 40~50% 정도의 계산량 감소가 있고, G.729A에서 SMV에서는 20% 내외의 계산량 감소가 있음을 확인할 수 있다.

2. 지연 시간

음성 통신 시스템의 총 지연 시간은 알고리즘(Algorithmic), 처리(Processing), 그리고 전송(Transmission) 지연 시간을 합하여 계산이 가능하다. 그러나 전송 지연은 통신망의 구조와 상태에 따라 달라지기 때문에 본 논문에서는 고려하지 않고 처리 지연의 경우, 앞에서 보였던 계산량과 중복되는 결과이기에 여기서는 알고리즘 지연 시간만을 살펴본다.

SMV에서 G.729A로의 상호부호화에서 Tandem 방식을 사용할 경우 알고리즘 지연 시간은 총 35ms이다. 이것은 SMV의 한 프레임의 길이 20ms에 SMV 부호화에서 사용되는 예측 구간 10ms, 그리고 복호화된 신호를 G.729A로 부호화할 때 사용되는 예측 구간 5ms가 더해진 것이다. 그러나 제안된 알고리즘을 사용할 경우, G.729A로 부호화할 때 사용되는 예측구간 5ms는 필요치 않다. 선형 예측 분석 과정이 없이 LSP를 직접 변환하기 때문이다. 따라서 이 경우 필요한 알고리즘 지연 시간은 30ms이고 제안된 알고리즘을 통해서 5ms의 지연 시간을 줄일 수 있었다.

G.729A에서 SMV로의 상호부호화에서는 Tandem 방식의 경우, 역시 마찬가지로 35ms의 지연 시간을 가진다. 이것은 SMV 한 프레임을 만들기 위한 G.729A의 두 프레임에서 20ms, 그리고 G.729A 부호화에서 5ms

의 예측 구간 그리고 복호화된 신호를 다시 SMV로 부호화할 때 필요한 예측 구간 10ms이다. 제안된 알고리즘의 경우, 최대 10ms의 지연 시간을 줄일 수 있다. 하지만, 여기 신호를 찾는 과정에서 어느 정도의 지연 시간을 필요로 하였기에, 음질의 향상을 위해서 5ms의 지연 시간을 두었다. 총 지연 시간은 30ms이고 5ms의 알고리즘 지연 시간 감소가 있다.

3. 객관적 음질 평가

여러 가지 객관적 음질 평가 모델 중 ITU P.862 Perceptual Evaluation of Speech Quality (PESQ)^[12]를 사용하여 제안된 상호부호화 알고리즘의 객관적음질을 평가하였다. <표 3>과 <표 4>는 Tandem 방식과 제안된 파라미터 직접 변환 방식의 PESQ 점수를 정리한 것이다. <표 3, 4>를 보면, SMV에서 G.729A로 가는 상호부호화기에서는 제안된 파라미터 직접 변환 방식의 PESQ 점수가 Tandem 방식보다 높다는 것을 확인

표 3. 객관적 음질 평가 결과(SMV → G.729A)
Table 3. Comparison of PESQ score(SMV → G.729A).

입력	Mode	Tandem	Proposed
남	0	3.461	3.554
	1	3.303	3.475
	2	3.219	3.423
	3	3.188	3.367
여	0	3.135	3.210
	1	3.019	3.168
	2	2.991	3.119
	3	2.995	3.108

표 4. 객관적 음질 평가 결과(G.729A → SMV)
Table 4. Comparison of PESQ score(G.729A → SMV).

입력	Mode	Tandem	Proposed
남	0	3.498	3.498
	1	3.445	3.395
	2	3.375	3.343
	3	3.368	3.342
여	0	3.233	3.227
	1	3.178	3.112
	2	3.136	3.057
	3	3.123	3.049

할 수 있다. 그리고, G.729A에서 SMV로 가는 상호부호화기에서는 Tandem 방식의 PESQ 점수와 비등하다.

4. G.729A에서 SMV로의 상호부호화에서 평균전송률 G.729A에서 SMV로의 상호부호화에서는 각 프레임마다 전송률을 결정해야 하는데 그 평균 전송률에 따라서 음질 결과가 다르게 나온다. 음질이 Tandem 방식과 비슷하더라도 전송률이 너무 높다면 결과를 신뢰할 수 없다. <표 5>는 Tandem에서와 제안된 알고리즘을 이용했을 경우의 평균 전송률을 나타낸 것이다.

<표 5>에서 확인할 수 있듯이 제안된 상호부호화기에서 평균 전송률이 Tandem한 경우의 결과와 많은 차이가 나지 않는다. Tandem한 경우에 비해서 평균 전송률이 비슷하거나 오히려 약간 낮은 결과를 보인다. 이 결과에서 전송률을 결정하는 알고리즘이 잘 동작한다는 것을 확인할 수 있다.

표 5. 평균 전송률(G.729A → SMV)

Table 5. Comparison of average bit rate (G.729A → SMV).

입력	Mode	Tandem	Proposed
남	0	5830.50	5082.75
	1	4413.38	4086.75
	2	3410.63	3374.25
	3	3135.38	3000.75
여	0	6231.88	5627.08
	1	4199.63	4390.89
	2	3470.03	3467.82
	3	3351.88	3293.54

VI. 결 론

본 논문은 G.729A와 SMV 음성부호화기를 위한 새로운 파라미터 직접 변환 방식의 상호부호화 알고리즘을 제안하였다. 제안된 알고리즘은 C 언어를 이용해서 구현되었고, 성능이 여러 가지 객관적 지표를 통하여 평가되었다. 제시된 결과는 제안된 알고리즘이 Tandem 방식에 비하여 더 효율적인 상호부호화를 할 수 있음을 보여준다. 제안된 알고리즘은 적은 계산량과 짧은 지연 시간으로도 동등한 수준의 음질을 보여주고 있다.

참 고 문 헌

- [1] ITU-T Rec. G.711, Pulse code modulation of voice frequencies, 1988.
- [2] ITU-T Rec. G.723.1, Dual-rate Speech Codec for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s, 1996.
- [3] ITU-T Rec. G.729 Annex A, Reduced Complexity 8 kbit/s CS-ACELP Speech Codec, 1996.
- [4] 3GPP TS 26.071 V5.0.0, AMR Speech CODEC: General Description, Jun., 2002.
- [5] TIA/ELA/IS-127, Enhanced variable rate codec, speech service option 3 for wideband spread spectrum digital systems, 1997.
- [6] ITU-T Rec. G.729, Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction(CD-ACELP), 1996.
- [7] 3GPP2 C.S0030-0 V2.0, Selectable Mode Vocoder Service Option for Wideband Spread Spectrum Communication System, 2001.
- [8] Yang Gao, E. Shlomot, A. Benyassine, J. Thyssen, Hyun-yu Su and C. Murgia, "The SMV Algorithm Selected by TIA and 3GPP2 for CDMA Applications", In Proc. ICASSP 2001, vol.2, pp 689-692, 1991.
- [9] Yang Gao, A. Benyassine, J. Thyssen, Hyun-yu Su and E. Shlomot, "EX-CELP : A Speech Coding Paradigm", In Proc. ICASSP 2001, vol.2, pp 689-692, 1991.
- [10] W.B. Kleijn, "Analysis-by-Synthesis Speech Coding Based on Relaxed Waveform-Matching Constraints," Ph.D. dissertation, Delft University of Technology, 1991.
- [11] Manfred R. Schroeder, Bishnu S. Atal, "Code Excited Linear Prediction(CELP): High-Quality Speech at Very Low Bit Rates," Proc. Of ICASSP, pp. 937-940, 1985.
- [12] ITU-T Rec. P.862, Perceptual evaluation of speech quality(PESQ), an objective method for

end-to-end speech quality assessment of narrowband telephone networks and speech codecs, 2000.

[13] F.K. Soong, B.H. Juang, "Line Spectrum Pair(LSP) and speech data compression," Proc. of ICASSP, pp. 1.10.1-1.10.4, 1984.

[14] Hong-Goo Kang, Hong-Kook Kim, R.V. Cox, "Improving transcoding capability of speech coders in clean and frame erased channel environments," Proc. of IEEE Workshop on Speech Coding, pp. 78-80, Jan., 2000.

[15] 이선일, 유창동, "AMR과 EVRC 음성 부호화기간의 비텐덤 방식을 이용한 상호 부호화," 한국음향학회지, 제 21권 제 6호, pp. 531-542, 2002

[16] Kyung Tae Kim, Sung Kyo Jung, Young Cheol Park, Yong Soo Choi, Dae Hee Youn, "An efficient transcoding algorithm for G.723.1 and EVRC speech coders," Proc. of 54th IEEE VTC, vol. 3, pp. 1561-1564, 2001.

[17] 윤성완, 정성교, 박영철, 윤대희, "8kbps G.729A에서 5.3 kbps G.723.1로의 상호 부호화 알고리즘," 신호처리 합동 학술대회 논문집 제 13권 제 1호, pp. 832-826, 2000

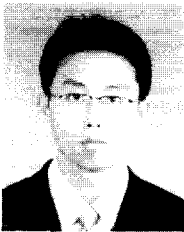
[18] Qualcomm Inc., CELP-based to CELP-based vocoder packet translation, US Patent no. US6260009B1, Jul., 2001.

저자 소개



張達源(正會員)
2002년 2월 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 학사. 2003년 8월 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 석사. 2003년 9월~현재 : 한국과학기술원 전자전산학과 전기 및

전자공학전공 박사과정



徐晟豪(正會員)
2002년 2월 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 학사. 2002년 9월~현재 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 석사과정

李善一(正會員)

2001년 2월 : 연세대학교 기계전자공학부 전자공학전공 학사. 2002년 8월 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 석사. 2002년 9월~현재 : 한국과학기술원 전자전산학과 전기 및 전자공학전공 박사과정

柳昌東(正會員)

1986년 : California Institute of Technology 학사. 1988년 : Cornell University 석사. 1996년 : MIT 박사. 1997년 1월~1999년 3월 : KT Senior Researcher. 1999년 3월~현재 : 한국과학기술원 전자전산학과 조교수