Transcoding Algorithm for G.723.1 and AMR Speech Coders: for Interoperability between VoIP and Mobile Networks[1]

Sung-Wan Yoon, Jin-Kyu Choi, Hong-Goo Kang, and Dae-Hee Youn

MCSP Lab. Dept. of Electrical and Electronic Eng., Yonsei University,
134 Shinchon-dong, Sudaemoon-gu, Seoul 120-749, Korea
yocello@mcsp.yonsei.ac.kr

Abstract

In this paper, an efficient transcoding algorithm between G.723.1 and AMR speech coders is proposed for providing interoperability between IP and mobile networks. Transcoding is completed through three processing steps: line spectral pair (LSP) conversion, pitch interval conversion, and fast adaptive-codebook search. For maintaining minimum distortion, sensitive parameters to quality such as adaptive and fixed-codebooks are re-estimated from synthesized target signals. To reduce overall complexity, other parameters are directly converted in parametric levels without running through the complete decoding process. Objective and subjective preference tests verify that the proposed transcoding algorithm has equivalent quality to conventional tandem approach. In addition, the proposed algorithm achieves 20~40% reduction of the overall complexity over tandem approach with a shorter processing delay.

1. Introduction

Among the many new speech coding standards, ITU-T G.723.1 [1] and ETSI adaptive multi rate (AMR) [2] cover wide range of applications requiring low bit rates. Both standards are used for different applications due to their distinctive features, but they obviously tend to share common applications such as digital cellular, voice messaging, and voice over internet protocol (VoIP). If the user of one network wants to successfully communicate with that of other network, interoperability is a crucial matter, so that endpoint devices are required to have a function to support both standard coders.

A simple approach to overcome this interoperability problem is to merge the decoder of one coder with the encoder of the other coder in cascade. However, the decoder-encoder tandem is often associated with several problems, such as quality degradation of the synthesized speech, computational complexity, and additional delay [3]. Unlike the tandem coding, *transcoding* can be used to overcome these difficulties [3]. Transcoding is a method to translate source bitstreams to target ones without running through complete decoding-encoding processes. Thus, it can minimize the quality degradation of the synthesized speech, computational complexity, and delay.

In this paper, we propose an efficient *transcoding* algorithm working between G.723.1 and AMR speech coders. According to the survey in [4], G.723.1 is the most widely deployed standards in VoIP systems and AMR is a standard for 3G mobile networks. Considering the frame length of two-speech coders, 30 ms to G.723.1 and 20ms to AMR, parameters corresponding to two frames of G.723.1 are converted to three frame sets of equivalent AMR parameters.

The proposed transcoding algorithm is composed of three processing steps: LSP conversion, pitch interval conversion, and fast adaptive-codebook search. Because the time interval to translate, 60 ms, is longer than that of pairs of [3], each part of transcoding should be modified in several aspects. In the LSP conversion, we may have flexibility because of using the LSP information of previous and current frame of source coder. In the pitch interval conversion, due to the same reason, the interval of pitch candidates covers the region around the pitch of source and target coder without the partial direct mapping used in [3]. Using the above three processing steps, the proposed transcoding algorithm translates bitstreams of source coder to those of target coder with minimum distortion and reduced complexity.

To verify the performance of the proposed algorithm, we perform objective and subjective tests such as perceptual evaluation of speech quality (PESQ) [8] with various speech sets. In addition, we measured the WMOPS [9] of each algorithm to compare the complexity of the algorithms. As a result, the proposed algorithm achieves 20~40% reduction of the overall complexity with a shorter processing delay while its quality is equivalent to tandem approach.

2. ITU-T G.723.1 and ETSI AMR

ITU-T G.723.1, standard for multimedia communication speech coder, has two modes whose bit rates are 5.3 and 6.3 kbps [1]. G.723.1 takes 30 ms of speech or other audio signals for encoding, and signals with the same length are reproduced by decoder. In addition, there is a look-ahead of 7.5 ms resulting in a total algorithmic delay of 37.5 ms.

AMR is the standard speech codec of 3GPP WCDMA. AMR encodes speech signal based on analysis-by-synthesis (AbS) algorithm. It has eight bit rates, from 4.75 to 12.2 kbit/s. The frame length of AMR is 20 ms, and each frame is divided by 4 subframes. For LPC analysis, 5 ms look-ahead is required, so total algorithmic delay is 25 ms. Main characteristics of two speech coders are summarized in Table 1.

Considering structures and parameters of the coder, shown in Table 1, the transcoding algorithms applied in each parameter are classified by two cases. The parameters having different structures and sensitive to quality are re-estimated. The parameters having same structures such as LPC parameters are directly converted in parametric domains.

[1] This work was supported in part by ETRI and BERC- KOSEF.

Table 1: Characteristic of G.723.1 and AMR

	G.723.1	AMR
Frame	30ms/frame	20ms/frame
LPC	10th order, LSP	10th order, LSP
ACB	5-tap predictor, integer resolution	1-tap predictor, 1/6 or 1/3 fractional
FCB	ACELP/MP-MLQ	ACELP

3. The Proposed Transcoding Algorithm

3.1. Transcoding from G.723.1 to AMR

The proposed transcoding algorithm has an asymmetric structure for Transmission (Tx) and Receive (Rx) paths. For the transmission of speech data from G.723.1 encoder to AMR decoder, transcoding process involves LSP conversion using linear interpolation and pitch interval conversion using pitch smoothing. Two frames of G.723.1 are translated to three frames AMR. A block diagram of the developed transcoding algorithm from G.723.1 encoder to AMR decoder is shown Figure 1.

LSP conversion using linear interpolation

A linear interpolation technique is employed to translate the LSP parameters. Given two sets of G.723.1 LSP, three frame sets of AMR LSP parameters are computed considering the frame length of two speech coders. Figure 2 shows the LSP conversion procedure, which can be denoted by

$$p_i^A(j)=\begin{cases} w_1\times p_0^G(j)+(1-w_1)\times p_1^G(j), & i=1 \\ w_2\times p_1^G(j)+(1-w_2)\times p_2^G(j), & i=2,\quad 1\le j\le 10 \\ p_2^G(j), & i=3 \end{cases} \qquad (1)$$

where p^G and p^A are the LSP parameters of G.723.1 and AMR, respectively, and i denotes the frame index of AMR. w_1 and w_2 are weighted values that are set considering geometrical distance. We set $w_1=0.33$ and $w_2=0.67$.

Considering the long time interval to translate, we also tested the efficiency of cubic spline interpolation technique. However, as shown in Table 2, there is no noticeable improvement nevertheless using the heavier computational load than the linear interpolation. To validate the efficiency of the proposed method, we compared the LPC spectrum of the tandem and transcoding to that of the original one. Figure 3 shows the LPC spectrum in the voiced region of speech signal, in which the LPC spectrum of G.723.1 is also shown as a reference. As shown in Figure 3, the LPC spectrum obtained after the LSP conversion given in Eq. (1) matches closely to the reference spectrum, especially in the low frequency region

Table 2: Comparison of Spectral Distortion

Method		SD [dB]	Outliers(%) 2-4dB	Outliers(%) >4dB
Tandem		3.02	64.22	16.18
Transcoding	Linear	1.59	20.10	0.00
	Cubic	1.64	22.30	0.49

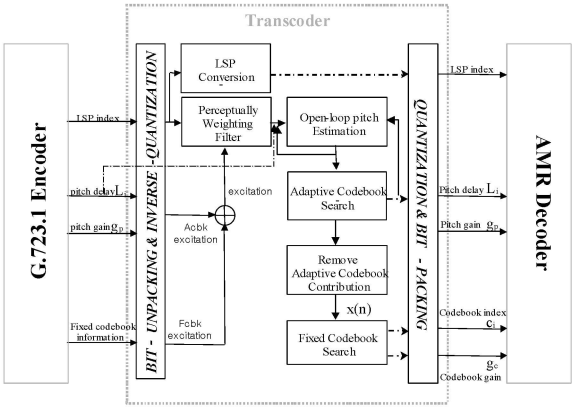


*Figure 1:*Structure of the transcoding from G.723.1 to AMR

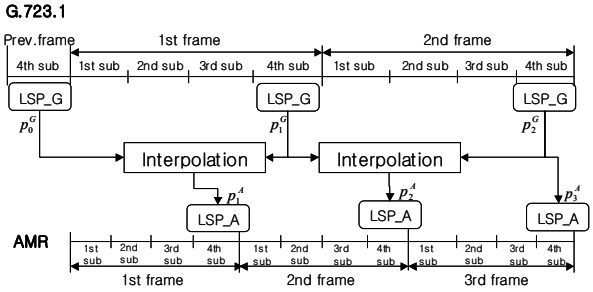


*Figure 2:*LSP conversion using linear interpolation

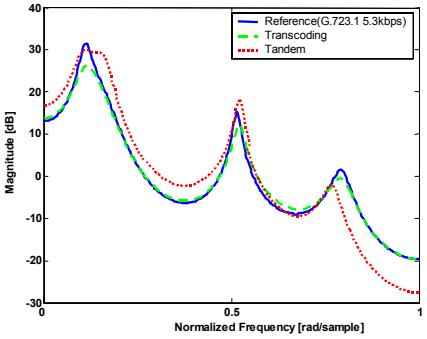


*Figure 3:*Comprison of LPC spectrum

and around the formant. The LPC spectrum after decoder-encoder tandem, however, indicates larger spectral distortion than the proposed LSP conversion. Since speech quality is mainly determined from the accuracy of low and formant frequency region components [5], it can be said that the proposed LSP conversion technique can provide better speech quality than the tandem. Since the proposed transcoding is not involved with the LPC analysis, the additional look-ahead delay is not required. Thus, the total delay of the proposed algorithm is at least 5 ms shorter than that of the tandem approach if we take similar approach to [6].

Pitch interval conversion using pitch smoothing

After the LSP conversion, the open-loop pitch for each frame of AMR is estimated. In the proposed transcoding

algorithm, the open-loop pitch is searched using a pitch smoothing technique. The interval for the pitch smoothing is set between the pitch value of the source and that of the target coder obtained in adjacent subframes. The search process is shown in Figure 4.

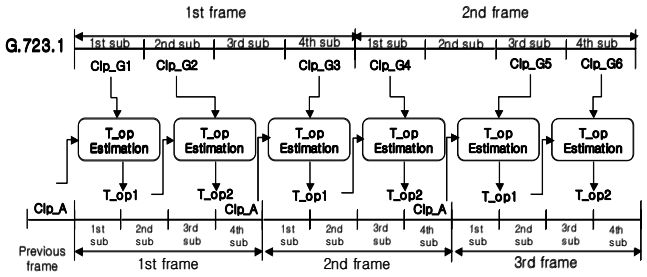


*Figure 4:*Pitch interval conversion using pitch smoothing

At first, the difference of the two candidates is computed to estimate the open-loop pitch of AMR:

$$\Delta = |p_G - p_A|, \tag{2}$$

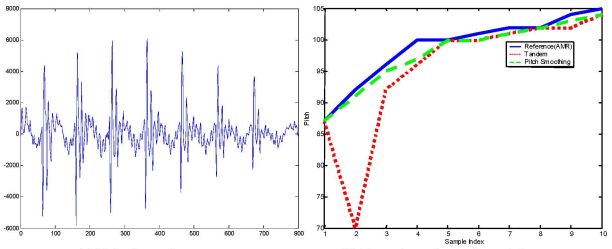
where p_G and p_A is the pitch of G.723.1 and AMR at adjacent subframe, respectively. Using the difference between two pitches, the constrained open-loop pitch search range, u , is determined such as :

$$\begin{cases} p_G - (\Delta/2) \le u \le p_G + (\Delta/2) \\ p_A - (\Delta/2) \le u \le p_A + (\Delta/2) \end{cases} \tag{3}$$

Among the constrained search ranges, the local optimum value maximizing the cost function of original open-loop pitch estimation of AMR encoding process is determined. And then, the two local maximum values in each part, G.723.1 and AMR, are compared each other in favor of AMR part. In other words, if the local maximum of AMR is larger than the 3/4 times of G.723.1, it is determined as the open-loop pitch. Otherwise, the local maximum of G.723.1 is selected. The value of 3/4 was determined by our large number of heuristic experiments.

To evaluate the performance of the pitch smoothing technique, we compare the estimated open-loop pitch contour of target coder with the adaptive-codebook pitch contour of source and target coders. As a result, the adaptive-codebook pitch contour of target coder well matches with the estimated open-loop pitch contour rather than the pitch value of source one. We inferred from the observation that the variation of pitch value between adjacent subframes are relatively stable and the estimated open-loop pitch are close to the previous closed-loop pitch value. As shown in Figure 5, the pitch contour of proposed method plotted by dashed line well matches with the original pitch value of AMR without any severe fluctuation. However, in the case of tandem approach, a drastic fluctuation appears like pitch multiple errors even in the stable voiced speech segment.

In the proposed scheme, the autocorrelation of the weighted speech is computed around the constrained region. In addition, as the search process to find the local maximum value in full search range is not involved, the open-loop pitch can be estimated with much less computational load. Furthermore, the pitch smoothing scheme can reduce the drastic fluctuation of pitch contour in voiced or voice-like segment, as shown in Figure 5, unlike that of tandem approach that has a chance to estimates inaccurate pitch information, which results in performance degradation of excitation modeling.



(a) Voiced speech segment (b) The estimated open-loop pitch contour

*Figure 5:*Comparison of open-loop pitch contour

3.2. Transcoding from AMR to G.723.1

Fast adaptive-codebook search

In G.723.1, the adaptive codebook search uses a 5th order pitch predictor. This process is computationally demanding because the pitch delay and pitch gain are searched simultaneously. In the transcoding algorithms, a fast adaptive codebook search algorithm [7] that pitch delay and pitch gain are computed sequentially is applied. This algorithm enables the system to significantly save a complexity [7].

The pitch gain G.723.1 coder is vector-quantized using 170 or 85-entry codebook. This process is another major computational burden for implementation. In the developed transcoding algorithm, the search range of gain codebook is limited depending on the pitch gain of AMR. In other words, the indices of the adaptive-codebook gain table are pre-selected depending on speech signal characteristics. Thus, the proposed approach considers the similarity or relationship of pitch gains of each coder. The process of gain index table generation or pre-selection is shown in Figure 6. The decoded PCM signal from AMR is encoded by G.723.1, like tandem connection. In this process, we can find the statistical information between the pitch gains of two coders. The dynamic range of pitch gain value of AMR is from 0 to 1.2 and we divide this pitch gain range into the 8 sub-sections and the conjugate structure of AMR gain codebook is considered for the boundary value of each sub-section. In the following encoding process of G.723.1, the selected adaptive-codebook gain table indices are stored at each subframe. As a result, the distribution of the most probable top 85 or 40 gain indices of pitch gain table for 5.3 kbps or 6.3 kbps, respectively, can be listed up at each pitch gain sub-section of AMR. For reliability of gain index distribution, we used speech signal recorded by female and male speakers, each sentence is 8 sec long, and total 96 sentences. Results of the subjective listening test confirmed that no degradation of speech quality was introduced with drastically reduced complexity by the fast adaptive-codebook search algorithm proposed in this paper.

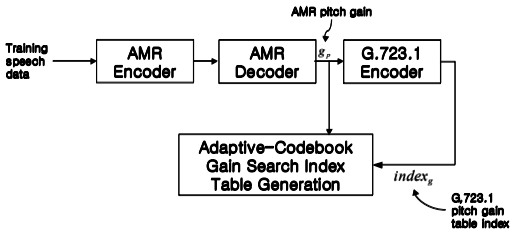


Figure 6:Gain index table generation for fast adaptive codebook

4. Performance Evaluations

4.1. Objective Quality Evaluation

For objective evaluation measures, NTT Korean speech database was used. Each sentence is 8 sec long, four male and female speakers and 12 sentences per each speaker are recorded under quiet environment, so total 96 sentences are used for PESQ evaluation.

As shown in Table 3, the proposed trancoding scheme is judged as comparable quality to tandem coding in general.

4.2. Subjective Quality Evaluation

For subjective evaluations informal A-B preference tests were conducted. The tests were performed by 30 naive listeners. In the tests, the subjects were asked to choose a favor sound between pairs of samples presented over headset. If the subjects could not distinguish the quality difference, they were asked to choose "no preference". The test material included 4 clean speech sentences obtained from two male and two female speakers. As shown in Table 4, tandem and transcoding were preferred in a similar ratio. Results implied that the listeners could not distinguish the quality of tandem coding from that of transcoding. Thus, we can conclude that the proposed transcoding algorithm produces equivalent quality to tandem coding.

4.3. Complexity

To check the complexity of the proposed algorithm, we computed the WMOPS of both tandem and transcoding algorithms. Results in Table 5 indicate that the processing time of each module, being compared with tandem coding, was noticeably reduced by using the transcoding algorithm. Also, the total encoding time of transcoding was close to 60-80% of the encoding time needed for tandem coding. Thus, it can be said that the developed transcoding algorithm can synthesize equivalent quality to the tandem coding with complexity about 20-40% lower than tandem coding.

5. Conclusion

In this paper, we proposed the transcoding algorithm that could convert G.723.1 bitstream into AMR bitstream, and vice versa. The proposed transcoding algorithm is composed of three steps: LSP conversion, pitch interval conversion, and fast adaptive-codebook search. Subjective and objective evaluation results showed that the proposed transcoding algorithm could produce equivalent speech quality to the tandem coding with shorter delay and less computational complexity.

Table 3: Objective test result[1]

Transcoding Direction	PESQ	
	Tandem	Transcoding
G.6.3k➔A.12.2k	3.488	3.523
G.6.3k ➔ A.7.4k	3.257	3.274
G.6.3k➔A.5.15k	3.081	3.008
A.12.2k➔G.6.3k	3.425	3.427
A.7.4k➔G.6.3k	3.311	3.355
A.5.15k➔G.6.3k	3.123	3.160

Table 4: Subjective test result

Transcoding Direction	ABX Preference(%)		
	Tandem	No Preference	Transcoding
G.6.3k➔A.12.2k	31	40	29
G.6.3k ➔ A.7.4k	38	27	35
G.6.3k➔A.5.15k	23	42	35
A.12.2k➔G.6.3k	24	29	47
A.7.4k➔G.6.3k	20	42	38
A.5.15k➔G.6.3k	28	30	42

Table 5: Comparison of complexity(WMOPS)

WMOPS	G.(6.3k) ➔ A.			A. ➔ G.(6.3k)		
AMR	12.2k	7.4k	5.15k	12.2k	7.4k	5.15k
Tandem	38.43	36.95	28.34	47.38	47.21	47.31
Transcoding	30.48	31.30	22.84	27.41	27.33	27.44
Reduction(%)	20.69	15.29	19.43	41.9	42.1	42.0

6. References

[1] ITU-T Rec. G.723.1 "Dual-rate Speech Coder For Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s," 1996.

[2] 3GPP TS 26.090 V5.0.0, AMR speech codec; Transcoding functions, Jun., 2002.

[3] S. W. Yoon, et al, "An Efficient Transcoding Algorithm for G.723.1 and G.729A Speech Coders," *Proc. Eurospeech 2001*, pp. 2499-2502, Sep., 2001

[4] O. Hersent, et al, *IP Telephony Packet-based multimedia communications systems*, Addison Wesley, 2000.

[5] L. R. Rabiner and R W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, 1978.

[6] H. G. Kang, et al, "Improving Transcoding Capability of Speech Coders in Clean and Frame Erasured Channel Environments," *Proc. IEEE Workshop on Speech Coding*, pp. 78-80, 2000.

[7] S. K. Jung, et al, "A Proposal of Fast Algorithms of ITU-T G.723.1 for Efficient Multi Channel Implementation," *Proc. Eurospeech 2001*, pp. 2017-2020, sep., 2001.

[8] ITU-T Rec. P.862 "Perceptual Evaluation of Speech Quality(PESQ), an Objective Method of End-to-End Speech Quality Assessment of Narrowband Telephone Networks and Speech Codec," May 2000.

[9] ITU-T Draft Rec. P.191 "Software tools for speech and audio coding standardization," Nov. 2000.

[1] Though we implemented all the possible combinations for transcoding of two coders, we only show the results from 3 bit-rates of AMR and higher bit-rate of G.723.1 because of lacking in space.