AN 8-12 KBIT/S EMBEDDED CELP CODER INTEROPERABLE WITH ITU-T G.729 CODER: FIRST STAGE OF THE NEW G.729.1 STANDARD

Dominique Massaloux[1], *Romain Trilling*[*], *Claude Lamblin*[1], *Stéphane Ragot*[1], *Hiroyuki Ehara*[2], *Mi Suk Lee*[3], *Do Young Kim*[3], *and Bruno Bessette*[4]

[1] France Télécom R&D, Lannion, France
[2] Matsushita Electric (Panasonic), Japan
[3] ETRI, Korea
[4] VoiceAge Corp., Canada

ABSTRACT

ITU-T G.729.1 is a scalable coder recently standardized in ITU-T for wideband telephony and voice over IP (VoIP) applications. Composed of three stages, this codec provides a scalable bitstream between 8 and 32 kbit/s both in narrowband and wideband. This paper describes the first stage which is a narrowband embedded CELP coder at 8 and 12 kbit/s. The 8 kbit/s layer ensures interoperability with ITU-T G.729 standard with a reduced complexity, and with a quality better than G.729 Annex A. At 12 kbit/s, G.729.1 reaches the quality level of the 11.8 kbit/s G.729 Annex E in spite of the embedded structure. The modifications brought to the original G.729 scheme to achieve this performance are explained and formal test results provided.

Index Terms— Speech coding, CELP, ITU

1. INTRODUCTION

Recently standardized by ITU-T, G.729.1 [1] is a new scalable speech and audio coder that extends ITU-T G.729 [2] speech coding standard widely used in voice over IP (VoIP) systems and is fully interoperable with it. G.729.1 will allow smooth transition from narrow band (300-3400 Hz) PSTN quality telephony to high quality wideband (50-7000Hz) telephony over IP and efficient deployment in existing infrastructures.

G.729.1 can operate at 12 bit rates from 32 kbit/s down to 8 kbit/s with wideband quality above 14kbit/s. The bit rate can be adjusted on the fly during a call by simple truncation of the embedded bitstream at any point of the communication chain such as gateways or other devices combining multiple data streams. Thanks to this highly flexible bit rate adaptation, optimum voice quality is provided according to service and network constraints and packet droppings that severely impair the overall quality are avoided.

Among many functionalities, this codec has a low delay mode at 8 and 12 kbit/s and can operate with input and/or output signals of 8 kHz sampling frequency. At 8 and 12 kbit/s, G.729.1 output has a bandwidth of 50-4000 Hz. Besides G.729 bitstream interoperability, G.729.1 offers enhanced narrowband quality close to toll quality at 12 kbit/s.

The G.729.1 algorithm is based on a three-stage embedded coding structure [3]: embedded Code-Excited Linear Predictive (CELP) coding of the lower band (50-4000 Hz), parametric coding of the higher band (4000-7000 Hz) by Time-Domain Bandwidth Extension (TDBWE), and enhancement of the full band (50-7000Hz) by predictive transform coding referred to as Time-Domain Aliasing Cancellation (TDAC).

This paper describes the 8-12 kbit/s CELP embedded scheme corresponding to the narrowband part of G.729.1. An overview of the complete G.729.1 structure can be found in [3]. Based on G.729, the 8 kbit/s core introduces modifications that allow both quality improvements and complexity reduction. This is presented in Section 2. Section 3 describes the 12 kbit/s embedded layer and Section 4 the post-processing for both layers. Finally, the performance (in terms of quality, complexity and delay) of the CELP stage is discussed in Section 5.

2. DESCRIPTION OF THE 8 KBIT/S LAYER

The 8 kbit/s core coder is derived from G.729 coder [2]. It uses the same fixed codebook and the same gain quantization as G.729. Modifications have been introduced to reduce phase distortion and complexity while improving quality. The high-pass elliptic pre-processing of G.729 is suppressed and the open-loop pitch estimation procedure is changed so that the pitch track is smoothed to improve frame erasure concealment (FEC).

The main modification lies in the fixed codebook (FCB) search. The FCB search is orthogonalized and a fast procedure is performed using global pulse replacement.

[*] *Romain Trilling was with France Telecom R&D, Lannion*

2.1 Orthogonalization of the FCB search

The fixed codebook search is performed using an orthogonal search procedure. In this procedure, fixed codebook vectors and the target vector, *x(n)*, used in the adaptive codevector search are orthogonalized to the filtered adaptive codevector, *y(n)*. The fixed codebook search is performed in a way that the distance between the orthogonal components of a filtered fixed codebook vector and the target vector is minimized. This search can be realized with two minor modifications to the original G.729 fixed codebook search as follows. Firstly, the target signal for the fixed codebook search, *x'(n)*, is calculated with:

$$x'(n) = \left(\sum_{i=0}^{39} y^2(i)\right)x(n) - \left(\sum_{i=0}^{39} x(i)y(i)\right)y(n), n = 0,\ldots,39,$$

Secondly, the conventional correlation matrix Φ of the impulse response $h(n)$ given by:

$$\Phi(i,j) = \left(\sum_{n=j}^{39} h(n-i)h(n-j)\right), i = 0,\ldots,39, j = i,\ldots,39$$

is changed to:

$$\Phi'(i,j) = \left(\sum_{n=0}^{39} y^2(n)\right)\Phi(i,j) - y'(i)y'(j)$$

where the correlation signal $y'(n)$ is obtained from the filtered adaptive codebook vector $y(n)$ and the impulse response $h(n)$ by:

$$y'(39-n) = \sum_{j=0}^{n} h(j)y(39-n+j), n = 0,\ldots,39$$

Using the above target vector $x'(n)$ and the matrix Φ', the search can be done in the same way as G.729, and its overhead in computational complexity is only the calculation of $y'(i)y'(j)$. However, this overhead does not become a problem, since a fast codebook search procedure is adopted in G.729.1, which is described below.

2.2 Fast codebook search using global pulse replacement

The fast search procedure only concerns the pulse positions. The pulse amplitudes (signs) are pre-selected using the G.729 technique. The fixed codebook, comprising four pulses, is searched based on global pulse replacement method [4]. This procedure consists of two stages: initial codevector determination and pulse replacement. The initial codevector is determined by the pulse positions which maximize in each track the pulse-position likelihood-estimate vector $b(n)$ given by:

$$b(n) = \frac{d(n)}{\sqrt{\sum_{i=0}^{39} d(i)d(i)}} + \frac{r_{LTP}(n)}{\sqrt{\sum_{i=0}^{39} r_{LTP}(i)r_{LTP}(i)}} \qquad n = 0,\ldots,39$$

where $r_{LTP}(n)$ is the long-term prediction residual signal, which is given by:

$$r_{LTP}(n) = r(n) - g_p v(n) \qquad n = 0,\ldots,39$$

where $r(n)$ is the LP residual signal, g_p is the pitch gain, and $v(n)$ is the adaptive codebook vector.

After the initial codevector determination, the optimum codevector is searched by replacing the initial pulse of each track with a new pulse which maximizes the search criterion C_k^2 / E_k. The pulse replacement procedure is repeated 4 times or terminated when C_k^2 / E_k does not increase any more. For example, let us assume that the initial codevector is determined as $(m_{01}, m_{11}, m_{21}, m_{31})$. In the symbol m_{xy}, x is the track number and y is the pulse position in each track. At the first step, a new codevector is searched by maximizing C_k^2 / E_k. If C_k^2 / E_k is maximized at codevector $(m_{03}, m_{11}, m_{21}, m_{31})$ and greater than that of initial codevector $(m_{01}, m_{11}, m_{21}, m_{31})$, then the codevector $(m_{03}, m_{11}, m_{21}, m_{31})$ is selected as a new codevector. As a result of the first step, m_{01} in the initial codevector is replaced with m_{03}. In the second step, the same procedure is repeated except for track 0. Then, a new codevector is selected in the same manner as that of the first step. As the pulse replacement procedure is repeated, the ratio C_k^2 / E_k of the replaced codevector is increased. This modification results in lower complexity and better quality by reducing the set of the pulse positions combinations.

3. DESCRIPTION OF THE 12 KBIT/S LAYER

The 12 kbit/s layer is based on a cascade CELP structure. It consists of adding an extra FCB to the core coder to enrich the CELP excitation. To offer an enhanced quality by addition of only 4 kbit/s, the extra codebook has been designed to better represent the highest frequencies of the lower band spectrum, and the search criterion has also been modified accordingly. Special care has been taken to keep the complexity low and re-use as much as possible of the G.729 routines.

3.1. Extra fixed codebook

The 12 kbit/s extra codebook is defined by a tri-pulse pattern $-\alpha z^{-1} + 1 - \alpha z$, with a central pulse of +1 and two side pulses with lower magnitude and opposite sign $-\alpha$. Each codevector is obtained by adding 4 occurrences of this pattern, scaled by a sign (±1) factor. The centres of the pattern occurrences occupy the same sets of positions as the G.729 fixed codebook pulses.

Due to the patterns shape, the extra codebook contains much more high frequency components than the G.729 fixed codebook, as shown in Figure 1 (comparing the

average power spectrum of the codevectors from both codebooks). This feature allows to reduce the smoothing effect that the CELP model tends to introduce, thus improving the output signal clarity.

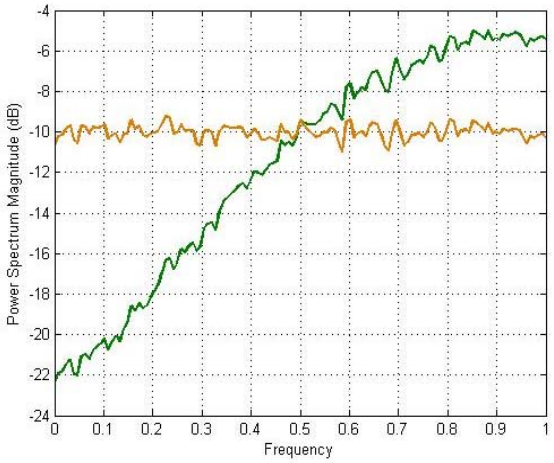

Figure 1. Average power spectrum for G.729 FCB codevectors (almost flat) and for 12 kbit/s FCB with α=0.34 (high-frequency content).

Note that the codevectors being defined by the positions of the pattern centres, the index is encoded using the same algorithm as G.729.

3.2 Tri-pulse pattern adaptation

The factor α is related to the nature of the signal and it is adapted to the amount of voicing. It has a value of 0 for purely unvoiced segments and of 0.34 for purely voiced segments. Therefore, the 12 kbit/s fixed codebook has more high frequency content in voiced signals and less for unvoiced signals. For each 5 ms subframe, α is given by:

$$\alpha = 0.17(1 + r_v)$$

where r_v is related to the voicing nature of the signal.

The value of r_v is given by

$$r_v = (E_v - E_c)/(E_v + E_c)$$

where E_v and E_c are the energies of the scaled pitch codevector and scaled innovation codevector, respectively.

3.3 Codebook search

First the perceptual weighting filter is modified to introduce a high pass emphasis so that the CELP criterion focuses on the higher frequencies of the narrow band: this is obtained by filtering the impulse response of the 1st layer perceptual filter with $-0.15z^{-1}+1-0.15z$.

Then, as described below, the choice of the G.729 pulse positions for the patterns centres positions has allowed to simplify the search and re-use the first-layer fast algorithm. The 12 kbit/s extra codebook samples can be expressed as:

$$C_2 = \begin{bmatrix} c_2(0) \\ c_2(1) \\ \vdots \\ c_2(38) \\ c_2(39) \end{bmatrix} = \begin{bmatrix} 1 & -\alpha & 0 & \cdots & 0 \\ -\alpha & 1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 & -\alpha \\ 0 & \cdots & 0 & -\alpha & 1 \end{bmatrix} \begin{bmatrix} c_1(0) \\ c_1(1) \\ \vdots \\ c_1(38) \\ c_1(39) \end{bmatrix}$$

where $C_1^T = (c_1(0), c_1(1), ..., c_1(39))$ describes the pattern centres position (same as G.729 fixed codebook pulse position), superscript $\cdot^T$ indicating transposition.

The perceptually filtered codevector involved in the CELP search is noted C_2^w. Filtering C_2 through a perceptual filter represented by a matrix:

$$H = \begin{bmatrix} h_0 & & & & \\ h_1 & h_0 & & 0 & \\ \vdots & \ddots & \ddots & & \\ h_{38} & & \ddots & \ddots & \\ h_{39} & h_{38} & \cdots & h_1 & h_0 \end{bmatrix}$$

gives the C_2^w. It can be shown that:

$$C_2^w = H \begin{bmatrix} 1 & -\alpha & 0 & \cdots & 0 \\ -\alpha & 1 & -\alpha & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -\alpha & 1 & -\alpha \\ 0 & \cdots & 0 & -\alpha & 1 \end{bmatrix} \begin{bmatrix} c_1(0) \\ c_1(1) \\ \vdots \\ c_1(38) \\ c_1(39) \end{bmatrix}$$
$$= H'C_1 + \alpha c_1(0)\bar{h}$$

where

$$H' = \begin{bmatrix} h'_0 & -\alpha h_0 & 0 & \cdots & 0 \\ h'_1 & h'_0 & -\alpha h_0 & \ddots & \vdots \\ \vdots & h'_1 & \ddots & \ddots & 0 \\ h'_{38} & \vdots & \ddots & h'_0 & -\alpha h_0 \\ h'_{39} & h'_{38} & \cdots & h'_1 & h'_0 \end{bmatrix}$$

$h'_i = -\alpha h_{i-1} + h_i - \alpha h_{i+1}$, $i = 0, \cdots, 39$ with $h_{-1} = 0$

and

$$\bar{h}^T = (h_1, h_2, \cdots, h_{40})$$

To simplify the codebook search, the term $\alpha c_1(0)\bar{h}$ can be neglected, provided that the value of α remains low enough. It has been verified that such a simplification does not degrade the performance in an audible manner. With this modification, the codebook search for the extra fixed codebook becomes quite similar to the G.729 search, allowing to re-use the fast process of the first layer.

3.4 Gain quantization

The extra fixed-codebook gains are scalar quantized relatively to the 8 kbit/s fixed codebook gain. Depending on the subframe index, the ratio of the two gains is scalar quantized on 3 bits or 2 bits, using a CELP error criterion as in G.729. Yet to avoid the introduction of low frequency artefacts, the perceptual weighting filter used to compute the gain does not include high frequency emphasis like that used for the codebook search.

4. NARROWBAND POST-PROCESSING

The lower-band adaptive postprocessing in G.729.1 comprises an adaptive postfilter derived from [2]. The parameters of the long-term and short-term postfilters depend on the decoder bit-rate. Moreover for 8 and 12 kbit/s output, the adaptive gain control is modified to attenuate fixed-point quantization errors in silence segments and a high-pass post-processing is also used.

5. PERFORMANCE

5.1. Subjective quality

G.729.1 quality was evaluated by formal ITU-T subjective tests [5]. Each experiment was conducted in two languages with 32 naive listeners using monaural headphones and 6 talkers (3 male and 3 female). G.729.1 coder met all requirements both in narrow and wideband conditions.

At 8 and 12 kbit/s, the quality test for clean speech signals at three input levels and 3% frame erasure was performed in French and North American English while the quality test on noisy speech signals was performed in German and Korean. In clean speech conditions, G.729.1 is better than G729 Annex A [6] at 8 kbit/s and equivalent to G.729 Annex E [7] at 12 kbit/s. On noisy speech signals, G.729.1 is better than G.729A at 12 kbit/s and equivalent to G.729A at 8 kbit/s (it is even better for two types of background noise –office and babble).

5.2 Delay and complexity

The algorithmic delay of G.729.1 is 48.9375 ms. However, for narrowband use cases, a low delay mode is provided bypassing the second and third G.729.1 stages. For narrow band input and output and bit rate limited to 8-12 kbit/s, the algorithmic delay is reduced to 25 ms.

The computational complexity of the embedded CELP coder (encoder + decoder) is 18.86 WMOPS at 8 kbit/s and 21.70 WMOPS at 12 kbit/s. These complexity figures correspond to the observed worst-case complexity of G.729.1 (using STL2005 v2.1 of ITU-T G.191A).

ACKNOWLEDGMENTS

Authors acknowledge the contribution of Yang Gao, Eyal Shlomot, Toshiyuki Morii, Jongmo Sung, Hyun-Woo Kim, and Martin Gartner in the development of the embedded CELP coder of ITU-T G.729.1.

REFERENCES

[1] ITU-T Rec. G.729.1, “An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729,” May 2006.

[2] ITU-T Rec. G.729, “Coding of Speech at 8 kbit/s using Conjugate Structure Algebraic Code Excited Linear Prediction (CS-ACELP),” March 1996.

[3] S. Ragot et al., “ITU-T G.729.1: An 8-32 kbit/s scalable coder interoperable with G.729 for wideband telephony and Voice over IP,” in Proc. ICASSP, 2007.

[4] E. D. Lee, M. S. Lee, and D.Y. Kim, “Global pulse replacement method for fixed codebook search of ACELP speech codec,” in Proc. IASTED CIIT, 2003.

[5] ITU-T SG16 Temporary Document, “LS on audio issues,” TD 202/GEN, ITU-T Q.10/16, Study Period 2005-2008, April 2006

[6] ITU-T Rec. G.729 Annex A, “Coding of Speech at 8 kbit/s using Conjugate Structure Algebraic Code Excited Linear Prediction (CS-ACELP): Reduced complexity 8 kbit/s CS-ACELP speech codec,” Nov. 1996.

[7] ITU-T Rec. G.729 Annex E, “Coding of Speech at 8 kbit/s using Conjugate Structure Algebraic Code Excited Linear Prediction (CS-ACELP): 11.8 kbit/s CS-ACELP speech coding algorithm,” Sept. 1998.