Abstract

This article describes the recently adopted ITU-T Recommendation G.729 Annex A (G.729A) for encoding speech signals at 8 kb/s with low complexity. G.729A is the standard speech coding algorithm for multimedia digital simultaneous voice and data (DSVD). G.729A is bitstream interoperable with G.729; that is, speech coded with G.729A can be decoded with G.729, and vice versa. Like G.729, it uses the conjugate-structure algebraic code excited linear prediction (CS-ACELP) algorithm with 10 ms frames. However, several algorithmic changes have been introduced which result in a 50 percent reduction in complexity. This article describes the algorithm introduced to achieve the low complexity goal while meeting the terms of reference. Subjective tests showed that the performance of G.729A is equivalent to both G.729 and G.726 at 32 kb/s in most operating conditions; however, it is slightly worse in the case of three tandems and in the presence of background noise. A breakdown of the complexities of both G.729 and G.729A is given at the end of the article.

ITU-T G.729 Annex A: Reduced Complexity 8 kb/s CS-ACELP Codec for Digital Simultaneous Voice and Data

Redwan Salami, Claude Laflamme, Bruno Bessette, and Jean-Pierre Adoul
University of Sherbrooke

Recently, there has been a great interest in multiplexing voice and data in multimedia terminals. At the request of Study Group 14 (SG 14) of the International Telecommunication Union — Telecommunication Standardization Sector (ITU-T), an expert group was established in February 1995 within SG 15 for the specification of a new speech coding standard for use in digital simultaneous voice and data (DSVD) applications [1]. The algorithmic complexity would be such that the modem algorithm (e.g., V.34) and speech coding algorithm could be implemented on the same processor (modem digital signal processor, DSP, or PC CPU). This was reflected in the terms of reference for the new algorithm, where an upper limit of 10 MIPS (million instructions per second) was imposed on the complexity. It was also required that the random access memory (RAM) not exceed 2000 words, and the read only memory (ROM) 8000 words. The main terms of reference are given in Table 1.

In summer 1995, the following five contending codecs were submitted to the host laboratory for subjective testing:

- Code-excited linear prediction (CELP)-based 7.73 kb/s with 15 ms speech frames from AT&T
- G.723.1-based 8.8 kb/s with 10 ms frames from Audio Codes/DSP Group (AC/DSPG)
- G.729-based 7.8 kb/s with 15 ms frames from NTT
- CELP-based 8 kb/s with 15 ms frames from Rockwell
- G.729 interoperable 8 kb/s with 10 ms frames from the University of Sherbrooke (USH)

The contending codecs were tested in both North American English and Japanese (at COMSAT and NTT). The test results were discussed at the September 1995 meeting of the expert group, where the codecs from AC/DSPG and USH came in ahead of the other codecs and were retained for further consideration. The codec from USH had the virtue of being bitstream interoperable with G.729. Because SG 15 felt that interoperability with G.729 would reduce the multiplicity of incompatible algorithms, the codec from USH was finally selected. The reduced-complexity version of G.729 for DSVD, described in Annex A of G.729, is now the standard speech codec in the ITU-T V.70 series (DSVD).

In this article, we first summarize the potential applications of this standard, and then describe the methods used to achieve the complexity reduction in the G.729 algorithm while maintaining a quality capable of meeting the terms of reference. The fast search methods applied to the pitch search and algebraic codebook search will be described, as well as the simplification of the postfiltering procedure. Subjective test results from the selection phase as well as the characterization phase will be given. Finally, a breakdown of the codec complexity of both G.729 and G.729A will be given.

Applications of G.729 Annex A

Although G.729 Annex A was specifically recommended by the ITU-T for multimedia DSVD applications, the use of the codec is not limited to these applications. In fact, due to its interoperability with G.729, G.729A can be used instead of G.729 when complexity reduction is deemed necessary in terminal equipment. Possible multimedia DSVD applications of G.729A are [2]:

- Multiparty multimedia conferencing (voice and data)
- Collaborative computing

0163-6804/97/$10.00 © 1997 IEEE

- Audiographic conferencing
- Telelearning and remote presentations
- Interactive games
- Multimedia bulletin boards and multimedia mail (voice and data)
- Telecommuting, teleshopping, and telemedicine
- File transfer during speech
- Automated teller machines with voice support
- Credit card verification
- Mobile audiovisual services
- Speech-to-text conversion

Another interesting potential application for G.729A is Internet telephony and Internet voice mail, where no standard speech coding algorithm exists. The relatively low complexity and low delay features of G.729A make it an attractive choice for such applications compared to G.723.1, the standard speech codec for public switched telephone network (PSTN) visual telephony (H.324), which has at least twice the complexity and three times the delay. In Internet applications, the low complexity feature of G.729A is important since the algorithm is likely to be run by the host processor in a window-based environment in which the processor will be performing other tasks simultaneously. The low delay feature becomes important in multiparty conferencing applications where more than one transcoding is needed.

Parameter	Requirement	Objective
Speech quality in error-free conditions	Not worse than that of G.726 at 32 kb/s	
Detected frame erasures — random (3% missing frames)	No more than 0.75 MOS degradation from G.726 at 32 kb/s	As small as possible
Speech quality dependency on the input signal level	Not worse than that of G.726 at 32 kb/s	As low as possible
Algorithmic delay	≤ 20 ms	≤ 10 ms
Total codec delay	≤ 40 ms	≤ 20 ms
Gross bit rate	< 11.4 kb/s	
Capability to transmit signaling/information tones	DTMF	As little distortion as possible
Tandeming capability for speech	2 asynchronous with a total distortion ≤ 4 asynchronous G.726	3 asynchronous ≤ 4 asynchronous G.726
Interoperation with voice activity detection	Required	
Complexity	≤ 10 MIPS ≤ 2000 words of RAM ≤ 8000 words of ROM	As low as possible
Performance in the presence of background noise	Not worse than 32 kb/s G.726	
Implementation	Fixed-point implementation	
Specification description	Bit-exact fixed-point modular ANSI-C code	

■ **Table 1.** *Main terms of reference for the DSVD speech codec. ANSI: American National Standards Institute.*

Parameter	Subframe 1	Subframe 2	Total
LSP coefficients			18
Pitch delay	8	5	13
Delay parity bit	1		1
Codebook positions index	13	13	26
Codebook signs index	4	4	8
Gains VQ (stage 1)	3	3	6
Gains VQ (stage 2)	4	4	8
Total			80

■ **Table 2.** *Bit allocation of the ITU-T 8 kb/s speech coder (G.729 and G.729A). VQ: vector quantization.*

Note that H.324 (PSTN videoconferencing) already includes codepoints for the use of G.729 or G.729A as an optional mode. Furthermore, since H.221, the transport mechanism for H.320 (integrated services digital network, ISDN, videoconferencing), requires the speech codec to operate at multiples of 8 kb/s, G.729A has the ideal rate for interoperability between V.70 (DSVD) and H.320. Hence, G.729A provides the additional benefit of more direct interoperability between V.70, H.320, and H.324, which are otherwise disparate multimedia recommendations [3].

General Description of the Coder

The general description of the coding/decoding algorithm of G.729A is similar to that of G.729 [4–7]. The same conjugate-structure algebraic code-excited linear-predictive (CS-ACELP) coding concept is used. The coder operates on speech frames of 10 ms corresponding to 80 samples at a sampling rate of 8000 samples/s. For every 10 ms frame, the speech signal is analyzed to extract the parameters of the CELP model (linear-prediction filter coefficients, adaptive and fixed codebook indices and gains). These parameters are encoded and transmitted. The bit allocation of the coder parameters is shown in Table 2. At the decoder, these parameters are used to retrieve the excitation and synthesis filter parameters. The speech is reconstructed by filtering this excitation through the short-term synthesis filter. The long-term or pitch synthesis filter is implemented using the so-called adaptive codebook approach. After computing the reconstructed speech, it is further enhanced by a postfilter. The encoding and decoding principles are further explained.

Encoder

The encoding principle is shown in Fig. 1. The input signal is high-pass filtered and scaled in the preprocessing block. The preprocessed signal serves as the input signal for all subsequent analysis. The 10th-order linear prediction (LP) analysis is done once per 10 ms frame to compute coefficients of the LP filter $1/A(z)$. These coefficients are converted to line spectrum pairs (LSPs) and quantized using predictive two-stage vector quantization (VQ) with 18 bits. The excitation signal is chosen by an analysis-by-synthesis search procedure. In this procedure, the error between the original and reconstructed speech is minimized according to a perceptually weighted distortion measure. This is done by filtering the error signal with a perceptual weighting filter, whose coefficients, unlike G.729, are derived from the quantized LP filter. A weighting filter of the form $W(z) = \hat{A}(z)/\hat{A}(z/\gamma)$ is used, where $\hat{A}(z)$ is the quantized version of $A(z)$.

The excitation parameters (fixed and adaptive codebook parameters) are determined for subframes of 5 ms (40 samples) each. The quantized LP filter coefficients are used for the second subframe, while interpolated LP filter coefficients are used in the first subframe. An open-loop pitch delay is estimated once per 10 ms frame based on the perceptually weighted and low-pass-filtered speech signal. Then the following operations are repeated for each subframe. The target signal $x(n)$ is computed by filtering the LP residual through the weighted synthesis filter $1/\hat{A}(z/\gamma)$. The initial states of this filter are updated by computing the weighted error signal at the end of the subframe. This is equivalent to the common approach of subtracting the zero-input response of the weighted synthesis filter from the weighted speech signal. The impulse response $h(n)$ of the weighted synthesis filter is computed. Closed-loop pitch analysis is then done (to find the adaptive-codebook delay and gain), using the target $x(n)$ and impulse response $h(n)$, by searching around the value of the open-loop pitch delay. A fractional pitch delay with 1/3 resolution is used. The pitch delay is encoded with 8 bits in the first subframe and differentially encoded with 5 bits in the second subframe. The target signal $x(n)$ is updated by subtracting the (filtered) adaptive-codebook contribution, and this new target, $x'(n)$, is used in the fixed codebook search to find the optimum excitation. An algebraic codebook with 17 bits is used for the fixed codebook excitation. The gains of the adaptive and fixed codebook contributions are vector quantized with 7 bits (with moving-average prediction applied to the fixed-codebook gain). Finally, the filter memories are updated using the determined excitation signal.

Decoder

The decoder principle is shown in Fig. 2. First, the parameter indices are extracted from the received bitstream. These indices are decoded to obtain the coder parameters corresponding to a 10 ms speech frame. These parameters are the LSP coefficients, the two fractional pitch delays, the two fixed codebook vectors, and the two sets of adaptive and fixed codebook gains. The LSP coefficients are interpolated and converted to LP filter coefficients for each subframe. Then, for each 5 ms subframe, the following steps are done:

- The excitation is constructed by adding the adaptive and fixed codebook vectors scaled by their respective gains.
- The speech is reconstructed by filtering the excitation through the LP synthesis filter.
- The reconstructed speech signal is passed through a post-processing stage. This includes an adaptive postfilter based on the long-term and short-term synthesis filters, followed by a high-pass filter and scaling operation.

Description of Algorithmic Changes to G.729

The LP analysis and quantization procedures as well as the joint quantization of the adaptive and fixed codebook gains are the same as G.729 [4–6]. The major algorithmic changes to G.729 are summarized below:

- The perceptual weighting filter uses the quantized LP filter parameters and is given by $W(z)= \hat{A}(z)/\hat{A}(z/\gamma)$ with a fixed value of $\gamma = 0.75$.
- Open-loop pitch analysis is simplified by using decimation while computing the correlations of the weighted speech.
- Computations of the impulse response of the weighted synthesis filter $W(z)/\hat{A}(z)$, of the target signal, and for updating the filter states are simplified by replacing $W(z)/A(z)$ by $1/\hat{A}(z/\gamma)$.

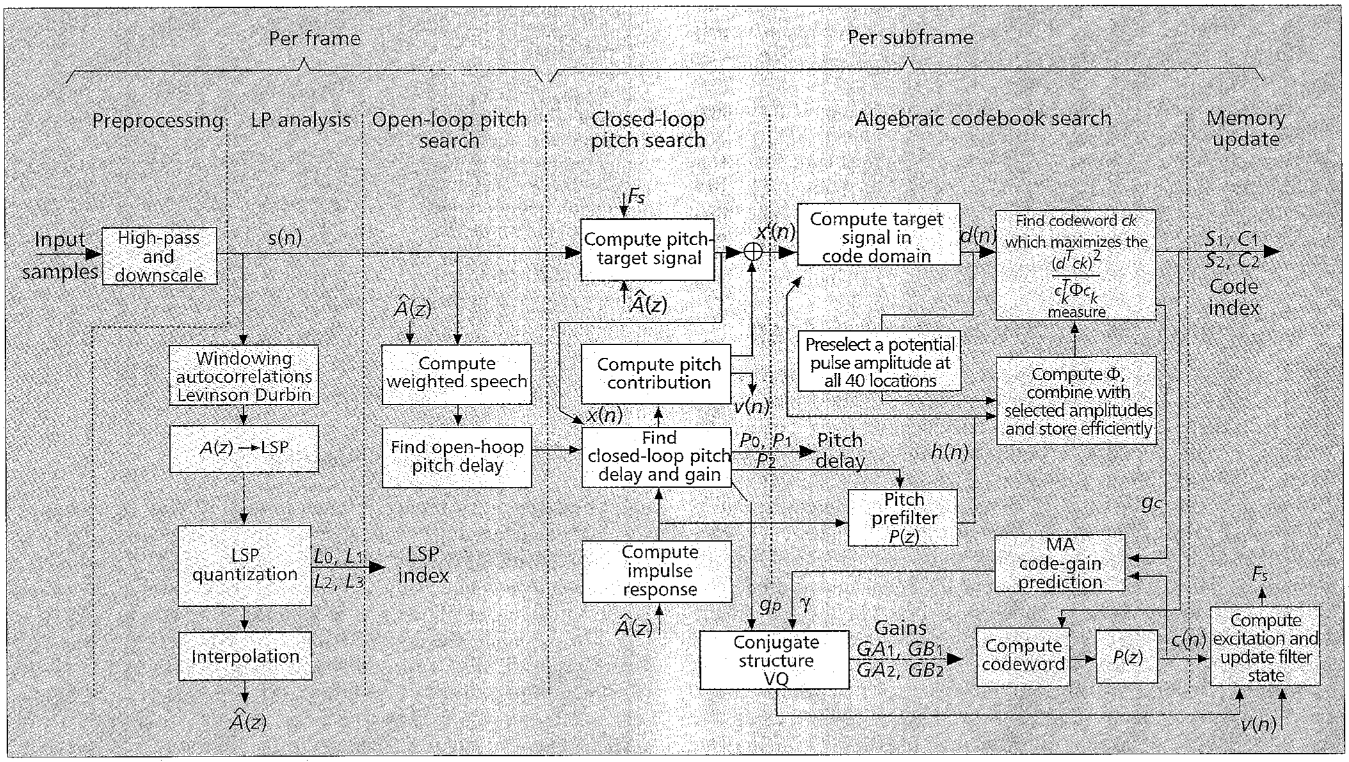
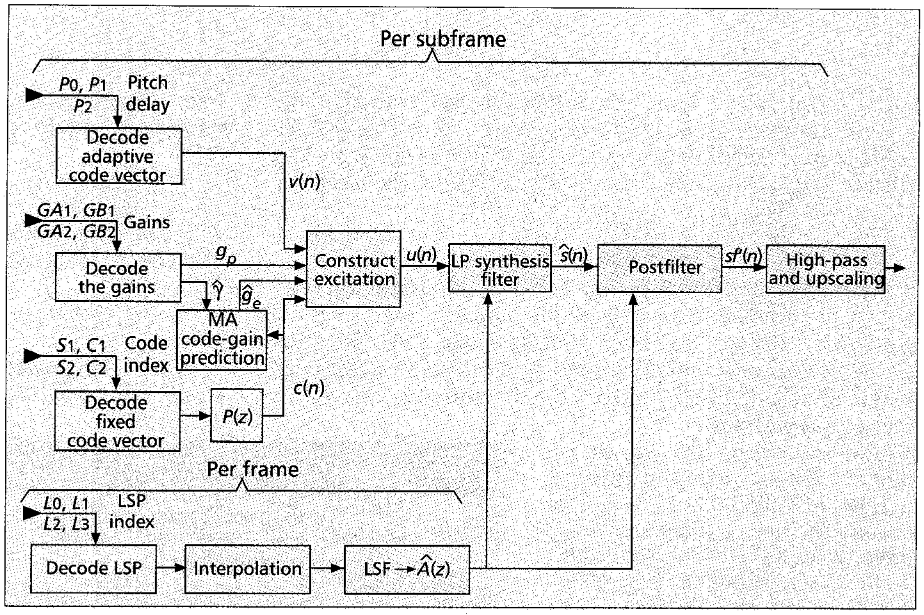

Per frame
Per subframe
Preprocessing
LP analysis
Open-loop pitch search
Closed-loop pitch search
Algebraic codebook search
Memory update
Input samples
High-pass and downscale
s(n)
Windowing autocorrelations Levinson Durbin
A(z) → LSP
LSP quantization
L0, L1
L2, L3
LSP index
Interpolation
Â(z)
Â(z)
Compute weighted speech
Find open-hoop pitch delay
Fs
Compute pitch-target signal
Â(z)
Compute pitch contribution
x(n)
v(n)
Find closed-loop pitch delay and gain
P0, P1
P2
Pitch delay
Compute impulse response
Â(z)
x'(n)
Compute target signal in code domain
d(n)
Find codeword ck which maximizes the (d^T ck)^2 / c_k^T Φ c_k measure
S1, C1
S2, C2
Code index
Preselect a potential pulse amplitude at all 40 locations
Compute Φ, combine with selected amplitudes and store efficiently
h(n)
Pitch prefilter P(z)
gc
MA code-gain prediction
gp
γ
Conjugate structure VQ
Gains
GA1, GB1
GA2, GB2
Compute codeword
P(z)
c(n)
Fs
Compute excitation and update filter state
v(n)

Figure 1. *Encoding principle of the CS-ACELP encoder in G.729 Annex A.*

- The adaptive codebook search is simplified. The search maximizes the correlation between the past excitation and the backward-filtered target signal (the energy of the filtered past excitation is not considered).
- The search of the fixed algebraic codebook is simplified. Instead of the nested-loop focused search, a depth-first tree search approach is used.
- At the decoder, the harmonic postfilter is simplified by using only integer delays.

These changes are described in more detail in the following sections.



■ **Figure 2.** *Principle of the CS-ACELP decoder in G.729 Annex A.*

Perceptual Weighting

Unlike G.729, the perceptual weighting filter is based on the quantized LP filter coefficients $\hat{a}_i$ and is given by

$$W(z)=\frac{\hat{A}(z)}{\hat{A}(z/\gamma)}, \tag{1}$$

with $\gamma = 0.75$. This simplifies the combination of synthesis and weighting filters to $W(z)/\hat{A}(z) = 1/\hat{A}(z/\gamma)$, which reduces the number of filtering operations for computing the impulse response and the target signal and for updating the filter states. Note that the value of γ is fixed to 0.75 and the procedure for the adaptation of the factors of the perceptual weighting filter described in G.729 [7] is not used in G.729A.

The simplification of the weighting filter resulted in some quality degradation in cases of input signals with flat response. In fact, the adaptation of the weighting factors was introduced in G.729 to improve the performance for such signals.

Open-Loop Pitch Analysis

To reduce the complexity of the search for the best adaptive codebook delay, the search range is restricted to a candidate delay T_{ol}, obtained from an open-loop pitch analysis. This open-loop pitch analysis is done once per frame (10 ms). The open-loop pitch estimation uses the low-pass-filtered weighted speech signal, $s_w(n)$, which is obtained by filtering the speech signal $s(n)$ through the filter $\hat{A}(z)/[\hat{A}(z/\gamma)(1 - 0.7z^{-1})]$. Open-loop pitch estimation is performed as follows. In the first step, three maxima of the correlation

$$R(k)=\sum_{n=0}^{39} s_w(2n)s_w(2n-k) \tag{2}$$

are found in the following three ranges:

$i = 1$: 20, ..., 39
$i = 2$: 40, ..., 79
$i = 3$: 80, ..., 143

The retained maxima $R(t_i)$, $i = 1, ..., 3$, are normalized through

$$R'(t_i)=\frac{R(t_i)}{\sqrt{\sum_{n=0}^{39} s_w^2(2n-t_i)}},\quad i=1,\ \ldots,\ 3. \tag{3}$$

The winner among the three normalized correlations is selected by favoring the delays with the values in the lower range. This is done by augmenting the normalized correlations corresponding to the lower delay range if their delays are submultiples of the delays in the higher delay range. The best open-loop delay T_{ol} is determined as follows:

```
if $|t_2 * 2 - t_3| < 5$
    $R'(t_2) = R'(t_2) + 0.25 * R'(t_3)$
if$|t_2 * 3 - t_3| < 7$
    $R'(t_2) = R'(t_2) + 0.25 * R'(t_3)$
if $|t_1 * 2 - t_2| < 5$
    $R'(t_1) = R'(t_1) + 0.20 * R'(t_2)$
if $|t_1 * 3 - t_2| < 7$
    $R'(t_1) = R'(t_1) + 0.20 * R'(t_2)$
$T_{ol} = t_1$
$R'(T_{ol}) = R'(t_1)$
if $R'(t_2) \geq R'(T_{ol})$
    $R'(T_{ol}) = R'(t_2)$
    $T_{ol} = t_2$
end
if $R'(t_3) \geq R'(T_{ol})$
    $R'(T_{ol}) = R'(t_3)$
    $T_{ol} = t_3$
end
```

Note that only half the number of samples is used in computing the correlations in Eq. 2. Furthermore, in the third delay region [80,143] only the correlations at the even delays are computed in the first pass; then the delays at ±1 of the selected even delay are tested.

Based on informal subjective tests, the simplification of the open-loop analysis did not introduce any significant degradation in the coder performance.

Closed-Loop Pitch Search

The adaptive codebook structure is the same as in G.729 [5, 8]. In the first subframe, a fractional pitch delay T_1 is used with a resolution of 1/3 in the range [19 1/3, 84 2/3] and integers only in the range [85, 143]. For the second subframe, a delay T_2 with a resolution of 1/3 is always used in the range $[int(T_1) - 5\ 2/3,\ int(T_1) + 4\ 2/3]$, where $int(T_1)$ is the integer part of the fractional pitch delay T_1 of the first subframe. This range is adapted for the cases where T_1 straddles the boundaries of the delay range.

Closed-loop pitch search is usually performed by maximizing the term

$$R(k)=\frac{\sum_{n=0}^{39} x(n)y_k(n)}{\sqrt{\sum_{n=0}^{39} y_k(n)y_k(n)}}, \tag{4}$$

where $x(n)$ is the target signal and $y_k(n)$ is the past filtered excitation at delay k (past excitation convolved with $h(n)$, the impulse response of the weighted synthesis filter $1/\hat{A}(z/\gamma)$). In this reduced complexity version, the search is simplified by considering only the numerator in Eq. 4. That is, the term

$$R_N(k)=\sum_{n=0}^{39} x(n)y_k(n)=\sum_{n=0}^{39} x_b(n)u_k(n) \qquad (5)$$

is maximized, where $x_b(n)$ is the backward filtered target signal (correlation between $x(n)$ and the impulse response $h(n)$) and $u_k(n)$ is the past excitation at delay k ($u(n-k)$). Note that the search range is limited around a preselected value, which is the open-loop pitch T_{ol} for the first subframe, and T_1 for the second subframe.

For the determination of T_2, and T_1 if the optimum integer delay is less than 85, the fractions around the optimum integer delay have to be tested. The fractional pitch search is done by interpolating the past excitation at fractions –1/3, 0, and 1/3, and selecting the fraction which maximizes the correlation in Eq. 5.

Simplifying the adaptive codebook search procedure resulted in some degradation compared to G.729. The chosen pitch lag occasionally differs by a fraction of 1/3 from that chosen in G.729.

Pulse no.	Track	Positions
0	T_0	0, 5, 10, 15, 20, 25, 30, 35
1	T_1	1, 6, 11, 16, 21, 26, 31, 36
2	T_2	2, 7, 12, 17, 22, 27, 32, 37
3	T_3	3, 8, 13, 18, 23, 28, 33, 38
	T_4	4, 9, 14, 19, 24, 29, 34, 39

■ **Table 3.** *Structure of the algebraic codebook.*

ALGEBRAIC CODEBOOK: STRUCTURE AND SEARCH

The structure of the 17-bit fixed codebook is the same as G.729 [5, 8]. The fixed codebook is based on an algebraic codebook structure using an interleaved single-pulse permutation design. The algebraic codebook is a deterministic codebook whereby the excitation code vector is derived from the transmitted codebook index (no need for codebook storage). In this codebook, each codebook vector contains four nonzero pulses. Each pulse can have either amplitude +1 or –1, and can assume the positions given in Table 3. The 40 positions in a subframe are divided into five tracks of eight positions each. The first three tracks can have one pulse each, while the last pulse is placed either in the fourth or fifth track. The sign of each pulse is quantized with 1 bit and its position is quantized with 3 bits, while 1 bit is used to determine whether the last pulse is placed in track T_3 or T_4. This gives a total of 17 bits.

The fixed codebook is searched by minimizing the mean squared error between the weighted input speech and the weighted reconstructed speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive-codebook contribution.

The matrix **H** is defined as the lower triangular Toepliz convolution matrix with diagonal $h(0)$ and lower diagonals $h(1),\ldots, h(39)$. The matrix $\Phi = \mathbf{H}^t\mathbf{H}$ contains the correlations of $h(n)$. If $\mathbf{c}_k$ is the kth fixed codebook vector, then the codebook is searched by maximizing the search criterion

$$\frac{C_k^2}{E_k}=\frac{(x^t\mathbf{H}\mathbf{c}_k)^2}{\mathbf{c}_k^t\Phi\mathbf{c}_k}=\frac{(\mathbf{d}^t\mathbf{c}_k)^2}{\mathbf{c}_k^t\Phi\mathbf{c}_k}, \qquad (6)$$

where $\mathbf{d} = \mathbf{H}^t\mathbf{x}$ is a vector containing the correlation between the target vector and the impulse response $h(n)$ (the backward filtered target vector) and t denotes transpose.

The vector **d** and the matrix Φ are computed before the codebook search. Note that only the elements actually needed are computed and an efficient storage procedure has been designed to speed up the search procedure.

The algebraic structure of the codebook C allows for a fast search procedure since the codebook vector $\mathbf{c}_k$ contains only four nonzero pulses. The correlation in the numerator of Eq. 6 for a given vector $\mathbf{c}_k$ is given by

$$C=\sum_{i=0}^{3} s_i d(m_i), \qquad (7)$$

where m_i is the position of the ith pulse and s_i is its amplitude. The energy in the denominator of Eq. 6 is given by

$$E=\sum_{i=0}^{3}\phi(m_i,m_i)+2\sum_{i=0}^{2}\sum_{j=i+1}^{3}s_is_j\phi(m_i,m_j). \qquad (8)$$

The search procedure is greatly speeded-up by the so-called *signal-selected pulse amplitude* approach. In this approach, the most likely amplitude of a pulse occurring at a certain position is estimated using $d(n)$ as side information. More precisely, the amplitude of a pulse at a certain position is set a priori equal to the sign of $d(n)$ at that position. To simplify the search procedure, the pulse amplitudes are predetermined by quantizing the signal $d(n)$, similar to G.729. This is done by setting the amplitude of a pulse at a certain position equal to the sign of $d(n)$ at that position. Therefore, before entering the codebook search, the following steps are taken. First, the signal $d(n)$ is decomposed into its absolute value $|d(n)|$ and its sign *sign* $[d(n)]$, which characterizes the preselected pulse amplitudes at each of the 40 possible pulse positions. Second, the matrix Φ is modified in order to include the preset pulse amplitudes; that is,

$$\phi'(i,j) = sign[d(i)]\; sign[d(j)]\; \phi(i,j), \quad i = 0,\ldots,39,\; j = i+1,\ldots,39. \qquad (9)$$

The main-diagonal elements of Φ are scaled to remove factor 2 in Eq. 8,

$$\phi'(i,i) = 0.5\phi(i,i),\; i = 0,\ldots,39. \qquad (10)$$

The correlation in Eq. 7 now reduces to

$$C = |d(m_0)| + |d(m_1)| + |d(m_2)| + |d(m_3)|, \qquad (11)$$

and the energy in Eq. 8 reduces to

$$\begin{aligned}E/2 &= \phi'(m_0,m_0)\\ &+ \phi'(m_1,m_1) + \phi'(m_0,m_1) \qquad (12)\\ &+ \phi'(m_2,m_2) + \phi'(m_0,m_2) + \phi'(m_1,m_2)\\ &+ \phi'(m_3,m_3) + \phi'(m_0,m_3) + \phi'(m_1,m_3) + \phi'(m_2,m_3).\end{aligned}$$

Having preset the pulse amplitudes, the next step is to determine the pulse positions that maximize the term C^2/E. In G.729, a fast search procedure based on a nested-loop search approach is used [5, 8, 9]. In that approach, only 1440 possible position combinations are tested in the worst case out of the 2^{13} position combinations (17.5 percent). In G.729A, in order to further speed up the search procedure, the search criterion C^2/E is tested for a smaller percentage of possible position combinations using a depth-first tree search approach. In this approach, the P excitation pulses in a subframe are partitioned into M subsets of N_m pulses. The search begins with subset 1 and proceeds with subsequent subsets according to a tree structure whereby subset m is searched at the mth level of the tree. The search is repeated by changing the order in which the pulses are assigned to the position tracks.

In this particular codebook structure the pulses are partitioned into two subsets ($M = 2$) of two pulses ($N_m = 2$). We begin with the following pulse assignment to tracks: pulse i_0 is assigned to track T_2, pulse i_1 to track T_3, pulse i_2 to track T_0, and pulse i_3 to track T_1. The search starts off with determining the pulse positions (i_0, i_1) by testing the search criterion for 2 x 8 position combinations (the positions at the two maxima of $|d(n)|$ in track T_2 are tested in combination with the eight positions in track T_3). Once the positions (i_0, i_1) are found, the search proceeds to determine the positions (i_2, i_3) by testing the search criterion for the 8 x 8 position combinations in tracks T_0 and T_1 (given pulses i_0 and i_1 are known). This gives a total of 16 + 64 = 80 combinations searched. This procedure is repeated by cyclically shifting the pulse assignment to the tracks; that is, pulse i_0 is now assigned to track T_3, pulse i_1 to track T_0, pulse i_2 to track T_1, and pulse i_3 to track T_2. The position combinations searched are now 2 x 80 = 160. The whole procedure is repeated twice by replacing track T_3 by T_4 since the fourth pulse can be placed in either T_3 or T_4. Thus, in total 320 position combinations are tested (3.9 percent of all possible position combinations).

Coder	Factor	MOS	Qeq
USH (8 kb/s)	– 16 dBov	3.61	26.53
	– 26 dBov	3.67	27.81
	– 36 dBov	3.52	24.92
	2 tandems	3.13	20.60
	3 tandems	2.51	16.08
G.726 (32 kb/s)	– 16 dBov	3.71	28.91
	– 26 dBov	3.59	26.07
	– 36 dBov	3.48	24.41
	4 tandems	2.64	16.93
Source	none	3.990	•
MNRU	Q = 30 dB	3.73	29.33
	Q = 24 dB	3.49	24.51
	Q = 18 dB	2.77	17.78
	Q = 12 dB	1.91	12.13

■ **Table 4.** *Test results of experiment 1 of the selection phase for the English language (performance in case of input level variations and tandems) — ACR method.*

About 50 percent of the complexity reduction in the coder part is attributed to the new algebraic codebook search (saving of about 5 MIPS). This was at the expense of slight degradation in coder performance (about 0.2 dB drop in signal-to-noise ratio, SNR).

Post-Processing

The post-processing is the same as in G.729 except for some simplifications in the adaptive postfilter. The adaptive postfilter is the cascade of three filters: a long-term postfilter,

$$H_p(z) = \frac{1}{1+\gamma_p gl}(1+\gamma_p glz^{-T});$$

a short-term postfilter,

$$H_f(z) = \frac{\hat{A}(z/\gamma_n)}{\hat{A}(z/\gamma_d)};$$

and a tilt compensation filter, $H_t(z) = 1 + \gamma_t k'_1 z^{-1}$, followed by an adaptive gain control procedure [4, 7]. Several changes have been undertaken in order to reduce the complexity of the postfilter. The main difference from G.729 is that the long-term delay T is always an integer delay and is computed by searching the range $[T_{cl} - 3, T_{cl} + 3]$, where T_{cl} is the integer part of the (transmitted) pitch delay in the current subframe bounded by $T_{cl} \leq 140$. The long-term delay and gain are computed from the residual signal $\hat{r}(n)$ obtained by filtering the speech $\hat{s}(n)$ through $\hat{A}(z/\gamma_n)$, which is the numerator of the short-term postfilter.

The modifications in the postfiltering procedure resulted in a reduction of about 1 MIPS in complexity.

Codec Subjective Performance

The DSVD codec performance was determined in two phases. In the so-called Selection Phase, the five original contenders were tested, resulting in the selection of a single codec. This codec was then submitted to a *Characterization Phase* of subjective testing. Because the coder was based on G.729, the tests used in this phase were less extensive than for the original G.729.

Selection Phase Results

In the Selection Phase, three experiments were performed on the contending codecs in both the Japanese and North American English languages, at NTT and COMSAT Laboratories, respectively. Experiment 1 dealt with the characterization of the test codecs with input-level variation and tandems (using modified IRS-weighted speech) [10]. Experiment 2 characterized the codec performance for clear speech and in the presence of burst frame erasures (using flat speech). Experiment 3 dealt with the performance of the contending codecs in the presence of background noise (babble noise at 20 dB SNR and a second talker at 15 dB SNR). In this article, only the results for the USH codec are given for the English language [11]. Note that the tested USH coder is the same as the final version of G.729A except for minor changes which were introduced to increase the common code between G.729 and G.729A.

In the COMSAT test design, the test material was obtained from six talkers (three males and three females) with six sentence pairs per talker. The number of listeners was 48 (six groups of eight listeners). In experiment 1, there were 36 test conditions, including six MNRU (modulated noise reference unit) conditions, where each condition received 288 votes. The listening devices used were monaural headphones. In the analysis of the test results, three statistical methods were used at 95 percent confidence level: Student's t-test Least Significance Difference (LSD), Tukey's Honestly Significant Difference (HSD), and Dunnet's Multiple Comparison method. More details about test conditions and analysis are found in [11].

Table 4 gives the subjective test results of experiment 1 (modified IRS-weighted speech) of the Selection Phase for the English language [11], with the absolute category rating (ACR) method [12]. The results are given in terms of mean opinion score (MOS) and equivalent Q (Qeq). The MNRU test conditions are used to derive a MOS vs. Q curve from which the Qeq value for each test condition is obtained [13]. From the statistical analysis of the results, the USH codec met all the requirements, and the objective for the 3 tandem condition [11].

Table 5 gives the subjective test results of experiment 2 (unweighted speech) of the Selection Phase for the English language [11] with the ACR method. From the statistical analysis of the results, the USH codec met the requirements for clear channel (equivalent to G.726) and for 3 percent frame erasure rate (less than 0.75 MOS degradation with respect to G.726 under error-free conditions). For the 5 percent forward error rate (FER), the codec was found statistically equivalent to 0.75 MOS degradation with respect to G.726 under error-free conditions.

Table 6 gives the subjective test results of Experiment 3 (unweighted speech) of the Selection Phase for the English language [11]. In this experiment, a five-point comparison category rating (CCR) method was used [12], with an MOS scale from –2 to 2.

From the statistical analysis of the results, the USH codec met the requirement for the interfering second talker but failed the requirement for babble noise.

Characterization Phase Results

The subjective tests for the Characterization Phase of G.729A were performed in May 1996 for both the Japanese and French languages at NTT and FT/CNET, respectively. The test consisted of three experiments [14]: Experiment 1 dealt with interworking between G.729 and G.729A (using an ACR method); experiment 2 dealt with the performance in the presence of background noise (using a CCR method); and experiment 3 dealt with the performance in the presence of channel errors and frame erasures (using an ACR method). Modified IRS weighted speech was used in all experiments. The results for the Japanese language are found in [15], and the conclusions of the three experiments are as follows. It was concluded from the results of experiment 1 that [15]:

- No significant difference was found among the four possible interconnections of G.729/G.729A and the reference coder (G.726 at 32 kb/s).
- The scores for all eight combinations with two-stage transcoding were higher than those for four-stage transcoding of G.726 at 32 kb/s.
- No significant difference was found between G.729A and G.726 at both high and low input levels.
- The quality of G.729A was slightly lower than that of G.729 under three-stage transcoding.

It was concluded from the results of experiment 2 that [15]:

- The scores of G.729A were slightly worse than those for G.729 and G.726 in both clear and background noise conditions.
- The scores using two-stage transcoding for both G.729A and G.729 were slightly worse than that for G.726 under background music conditions, although the differences were not significant for noise-free background and background office noise conditions.
- No significant differences were found for the possible combinations of the two-stage transcoding of G.729 and G.729A under noise-free and background office noise conditions.

It should be noted that the CCR assessment method used in experiment 2 is very good for exposing small differences in quality; however, this method does not necessarily reflect the user's assessment in the application field [15].

In experiment 3, G.729A and G.729 were tested with random bit errors at a rate of 10^{-3}, and 3 and 5 percent random frame erasures, in a quiet background, as well as in babble and office background noise conditions. In general, no statistically significant difference was found between G.729A, G.729, and their interconnections.

Coder	Factor	MOS	Qeq
USH (8 kb/s)	0 % FER	3.76	31.86
	3 % FER	3.18	26.61
	5 % FER	2.84	23.84
G.726 (32 kb/s)	0 % FER	3.65	30.74
Source	None	4.38	40.65
MNRU	Q30	3.59	30.21
	Q24	2.81	23.57
	Q18	2.20	18.52
	Q12	1.56	11.95

■ **Table 5.** *Test results of experiment 2 of the Selection Phase for the English language (performance in clear conditions and burst frame erasures) — ACR method.*

Coder	Factor	CMOS
USH (8 kb/s)	Babble	–0.09
	Sec. Talker	0.26
G.726 (32 kb/s)	Babble	0.12
	Sec. Talker	0.10
MNRU	Q30	0.17
	Q24	–0.17
	Q18	–0.93
	Q12	–1.57

■ **Table 6.** *Test results of experiment 3 of the Selection Phase for the English language (performance in the presence of background noise) — CCR method.*

Codec Implementation and Complexity

The reduced-complexity CS-ACELP codec in G.729 Annex A specification consists of 16-bit fixed-point ANSI C code using the same set of fixed-point basic operators used to define G.729. A set of test vectors are provided as part of G.729A to ensure that a certain DSP implementation is bit-exact with the fixed-point ANSI C code using basic operators. Basic operators are a C-language implementation of commonly found fixed-point DSP assembly instructions. Describing an algorithm in terms of basic operators allows for easy mapping of the C-code to a certain DSP assembly language as well as a rough estimate of the algorithmic complexity. A certain weight is associated with each basic operator which reflects the number of instruction cycles. Using these basic operators, the codec complexity was found to be 8.95 WMOPS (weighted million operations per second). A factor of 1.2–1.5 is usually used to estimate the complexity in MIPS (this depends on the DSP used and the actual function performed).

Both G.729A and G.729 were implemented on the TI TMS320C50 DSP chip. In the USH implementation, the full-duplex codec algorithm of G.729A required 12.4 MIPS, while that of G.729 required 22.3 MIPS. The breakdown of the complexity of both G.729A and G.729 is given in Table 7, for both encoder and decoder. The complexity is given in terms of C50 MIPS and basic operator's WMOPS. In terms of memory occupation, G.729A required less than 2K RAM and 10K ROM while G.729 required about 2K RAM and 11K ROM. It is evident that using G.729 Annex A, about 50 percent reduction in the complexity of G.729 is achieved, with a slight penalty represented by some degradation in performance in the case of three-stage transcoding and in the presence of background noise.

Conclusion

This article described the speech coding algorithm of Recommendation G.729 Annex A, which is the standard codec for multimedia digital simultaneous voice and data. This algorithm is bitstream interoperable with the algorithm specified in the main body of Recommendation G.729. It is an 8 kb/s algorithm based on the CS-ACELP coding concept, and uses 10 ms speech frames. This algorithm resulted in about a 50 percent reduction in the complexity of G.729 at the expense of small degradation in performance in the case of three tandems and in the presence of background noise.

More recently, a robust voice activity detection/comfort noise generation (VAD/CNG) procedure was adopted for G.729A in DSVD terminals in Annex B of G.729 [16]. This procedure uses discontinuous transmission (DTX) in case of background

Function	WMOPS		C50 MIPS	
	G.729	G.729A	G.729	G.729A
Pre-processing	0.20	0.20	0.226	0.226
LP analysis	1.63	1.28	1.957	1.696
LSP quantization & inter.	0.95	0.95	1.390	1.390
LSP to A(z) and weighting	0.30	0.12	0.461	0.173
Open-loop pitch	1.45	0.82	1.563	0.955
Closed-loop pitch	2.83	1.55	3.453	1.778
Algebraic codebook	6.35	1.86	8.406	3.046
Quantization of gains	0.46	0.46	0.643	0.643
Find exc. & memory update	0.21	0.08	0.278	0.112
Total (coder)	14.38	7.32	18.377	10.019
Decoder	0.68	0.68	1.133	1.133
Postfilter	2.13	0.73	2.539	1.000
Post-processing	0.22	0.22	0.266	0.266
Total (decoder)	3.03	1.63	3.938	2.399
Total (duplex)	17.41	8.95	22.315	12.418

■ **Table 7.** *Breakdown of the codec complexity (worst case) for G.729 and G.729A in terms WMOPS and TMS320C50 MIPS.*

noise where the 16 bits per 10 ms frame used to describe the spectrum of the background noise are only transmitted if a change in the background noise characteristics is detected. This robust VAD/DTX/CNG procedure resulted in about a 50 percent drop in the average bit rate in normal two-way conversations without affecting the codec performance. Limited subjective tests were performed and it was found that including the VAD/CNG procedure did not result in any degradation in the speech quality for several types of background noise.

Currently, ITU-T SG 16 is considering two bit rate extensions for G.729. The first extension is at 12 kb/s, and aims at improving the performance of G.729 for music signals and in the presence of background noise. The second bit rate extension is at 6.4 kb/s, to give G.729 the flexibility to lower the bit rate in case of network congestion. Floating-point versions of both G.729 and G.729A are also foreseen in the future.

With Annexes A and B of G.729 being finalized, and with its future bit rate extensions, G.729 and its Annexes become a complete speech coding package suitable for a wide range of applications in wireless, wireline, and satellite communications networks as well as Internet and multimedia terminals.

Acknowledgments

The authors would like to thank Simão F. de Campos Neto, Bastiaan Kleijn, and Peter Kroon for proofreading an early version of this article and for their valuable comments. The authors would also like to acknowledge the financial support of Sipro Lab Telecom to parts of this work.

References

[1] R. V. Cox, "Three New Speech Codecs from the ITU Cover a Range of Applications," *IEEE Commun. Mag.*, this issue.

[2] ITU-T SG 14 cont., "Liaison to Study Group 15 on G.DSVD," Source: SG 14, Apr. 1995.

[3] ITU-T SG 15 cont. DSVD-95-52, "G.DSVD and Interoperable Multimedia Standards," PictureTel Corp., Oct. 1995.

[4] ITU-T Draft Rec. G.729, "Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic Code-Excited Linear-Prediction (CS-ACELP)," 1995.

[5] R. Salami *et al.*, "Description of the Proposed ITU-T 8 kbit/s Speech Coding Standard," *Proc. IEEE Speech Coding Wksp.*, Annapolis, MD, Sept. 1995, pp. 3–4.

[6] A. Kataoka *et al.*, "LSP and Gain Quantization for the Proposed ITU-T 8 kbit/s Speech Coding Standard," *IEEE Speech Coding Wksp.*, Annapolis, MD, Sept. 1995, pp. 7–8.

[7] D. Massaloux and S. Proust, "Spectral Shaping in the Proposed ITU-T 8 kbit/s Speech Coding Standard," *IEEE Speech Coding Wksp.*, Annapolis, MD, Sept. 1995, pp. 9–10.

[8] R. Salami *et al.*, "Design and Description of CS-ACELP: A Toll Quality 8 kbit/s Speech Coder," To be published, *IEEE Trans. Speech and Audio Proc.*

[9] R. Salami *et al.*, "A Toll Quality 8 kb/s Speech Codec for the Personal Communications System (PCS)," *IEEE Trans. Vehic. Tech.*, vol. 43, no. 3, Aug. 1994, pp. 808–16.

[10] ITU-T Rec. P.48, "Specification for an Intermediate Reference System," vol. V, *Blue Book*, Geneva, Switzerland, 1989, pp. 81–86.

[11] ITU-T SG 15 cont., "Final Test Report of DSVD Experiments 1, 2 and 3 for North-American English," COMSAT, Geneva, Switzerland, Nov. 1995.

[12] ITU-T Rec. P.800, "Methods for Subjective Determination of Transmission Quality," Geneva, Switzerland, May 1996.

[13] P. Kroon, "Evaluation of Speech Coders," *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds., Elsevier, 1995.

[14] SQEG cont. SQ-35.96, "Subjective Test Plan for Characterization of an 8 kbit/s Speech Codec for DSVD Applications," ITU-T SG 12, Mar. 1996.

[15] ITU-T SG 15 cont., "Results of Characterization Testing Using Japanese Language for Draft Annex A to Recommendation G.729 (Low-Complexity CS-ACELP for DSVD Applications)," NTT, May 1996.

[16] A. Benyassine *et al.*, "ITU-T G.729 Annex B: A Silence Compression Scheme for Use with G.729 Optimized for V.70 DSVD Applications," *IEEE Commun. Mag.*, this issue.

Biographies

REDWAN SALAMI received the B.Sc. degree in electrical engineering from Al-Fateh University, Tripoli, Libya, in 1984, and the M.Sc. and Ph.D. degrees in electronics from the University of Southampton, U.K., in 1987 and 1990, respectively. In 1990, he joined the Department of Electrical Engineering, University of Sherbrooke, Quebec, Canada, where he is currently an adjunct professor involved in the design and real-time implementation of low-bit-rate speech coding algorithms. He contributed to several speech coding standards in ITU-T and the cellular industry, including ITU-T Recommendations G.729 and G.729 Annex A, 12.2 kb/s enhanced full-rate (EFR) GSM codec, and 7.4 kb/s EFR TDMA codec (IS-641). His research interests include speech coding, digital communications, and digital mobile radio systems.

CLAUDE LAFLAMME received the B.S. degree in electrical engineering from the University of Sherbrooke, Quebec, Canada, in 1984. Since 1985 he has been with the Department of Electrical Engineering, University of Sherbrooke, working on DSP implementation and design of speech coding algorithms. He is currently a senior researcher in the Information, Signal and Computer research group. His research interests are in digital speech coding and DSP development systems.

BRUNO BESSETTE received the B.S. degree in electrical engineering from the University of Sherbrooke, Quebec, Canada, in 1992. In 1993, he worked for SMIS (an R&D company) as software engineer and developed a teletex receiver for the account of HydroQuebec. In 1994, he joined the Electrical Engineering Department of the University of Sherbrooke, where he is currently a software engineer and researcher with the speech coding group. He has taken part in the design and real-time implementation of speech coding algorithms, many of them are currently standardized in the world.

JEAN-PIERRE ADOUL received the Diplome d'Ingenieur ENREA from the Ecole Nationale Superieur d'Electronique, France, in 1967, and the M.S. and Ph.D. degrees in electrical engineering from Lehigh University, Bethlehem, Pennsylvania, in 1968 and 1970, respectively. He was awarded a Fulbright scholarship to pursue graduate studies at Lehigh University. Since 1970 he has been on the faculty of applied sciences in the Department of Electrical Engineering at the University of Sherbrooke, Quebec, Canada, where he is a full professor teaching signal processing and pattern recognition, and head of the Information, Signal and Computer research group. He was a visiting associate professor at Stanford University, California, in spring 1978. He has conducted research in the area of channel modeling and in digital telephony, digital speech interpolation, speech coding, and detection.