

FAST PITCH MODELLING FOR CS-ACELP CODER USING FERMAT NUMBER TRANSFORMS

G. Madre^{†‡}, E.H. Baghious[†], S. Azou[†] and G. Burel[†]

[†]Laboratoire d'Electronique et Systèmes de Télécommunications - UMR CNRS 6165
6, avenue Le Gorgeu - BP 809 - 29285 BREST cedex - FRANCE

[‡]Société NETLINE – <http://www.netline.fr>

{madre,baghious}@univ-brest.fr

Abstract

This paper presents a double improvements to reduce the speech coding complexity of the pitch prediction in a Code-Excited Linear Prediction (*CELP*) coder. First, the pitch analysis structure is modified. A new fast Pitch Modelling by linear-Filtering (*PMF*) procedure will determine the adaptive and stochastic codebook contributions of the excitation signal. Afterwards, an efficient implementation of the *PMF – CELP* coding is proposed by using Number Theoretic Transforms which can significantly reduce the algorithm computation complexity.

1. Introduction

Most modern speech coding techniques are based on the Code-Excited Linear Prediction (*CELP*) paradigm due to its simplicity and high performances. This technique codes the speech signal with a Linear Prediction analysis and a pitch modelling. These analysis operations determine an excitation which is used to synthesize the speech signal [1].

According to the voiced and unvoiced speech features, we propose a new Pitch Modelling procedure using linear-Filtering (*PMF*), which splits the excitation signal up into predictable and unpredictable frames, to reduce the transmission rate and the computation time of the pitch analysis which represents a significant part of the coder complexity.

In a second stage, the *PMF – CELP* coding has been developed and implemented in fixed-point arithmetic with Number Theoretic Transforms (*NTT*), which offer many advantages over the Discrete Fourier Transforms [2] :

- Few or no multiplications are required
- Use of floating point complex numbers is removed and error-free computations are allowed
- Computations are executed on a finite ring of integers, which allows an efficient implementation into *DSP*

Hence, the use of Number Theoretic Transforms will reduce the delay features, by minimizing the computation complexity. The case of Fermat Number Transforms (*FNT*), with arithmetic carried out modulo a Fermat number, is particularly suited for digital convolution computations.

To illustrate the real benefits of an *FNT*-based implementation, the *PMF* procedure has been tested through numerical simulations of the *G.729* codec. The *G.729* recommendation of the International Telecommunication Union (*ITU*) standardizes the *8kb/s* Conjugate Structure - Algebraic *CELP* speech coding, which is targeted for digital simultaneous voice and data applications [3] [4].

The rest of the paper is organized as follows. Section 2 introduces the *CS – ACELP G.729* coder and focuses on the pitch prediction. In a third part, the new *PMF* procedure, adapted to the constraints of the *G.729* norm, is presented. Section 4 presents the concept of the Number Theoretic Transform and details the Fermat Number Transform, which will be implemented into the *PMF – CELP* coding. In the final part, the synthesized speech quality is evaluated and the benefits of the proposed *FNT*-based implementation are revealed through the *G.729* codec.

2. G.729 Pitch Prediction

The coder operates on speech frames of *10ms*, which correspond to 80 samples at a sampling rate of *8kHz* [3]. The speech signal is analyzed every frame to extract the 10^{th} order Linear Prediction (*LP*) filter coefficients, which are converted to Line Spectral Frequencies and quantized. Afterwards, the excitation parameters (pitch delay, adaptive and fixed codebook index and gains) are estimated on a 40-samples (*5ms*) subframe basis.

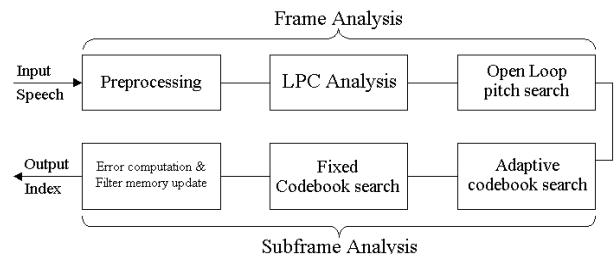


Figure 2.1 : Principal blocks of the *G.729* coder

The speech signal is approximated by an excitation signal processed by the synthesis filter [5]. The optimal excitation u is then constructed, for each subframe, as a linear combination of an adaptive and fixed codebooks contribution $\{v, c\}$, which model the predictable and stochastic parts of the excitation respectively (figure 2.2) :

$$u(n) = \beta v(n) + Gc(n) \quad (1)$$

where $n = 0, \dots, N - 1$, N is the subframe length and $\{\beta, G\}$ the gain factors of both codebook contributions.

In the *G.729* coder, the voiced components are selected from the adaptive codebook, which contains the past excitation, with a pitch analysis using a two-stages procedure. An *open-loop* pitch search estimates an interval of pitch values in which a *closed-loop* pitch analysis selects the optimale adaptive codebook contribution v .

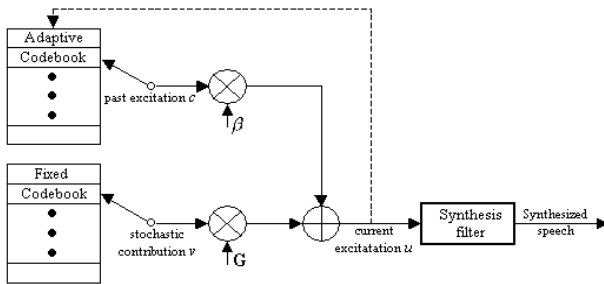
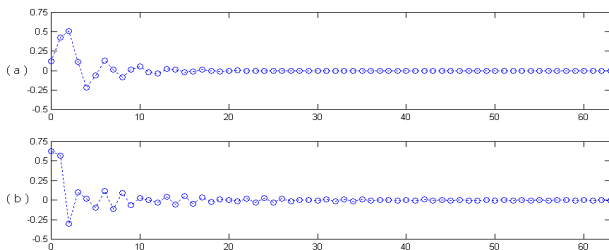


Figure 2.2 : CELP synthesis

3. PMF-CELP Coding

According to the voiced and unvoiced speech features, the codebook contributions are more or less significant [1]. However, the CELP coders do not proceed differently to code the excitation parameters. The following new Pitch Modelling by linear-Filtering divides basically the speech frames coding as the excitation is predictable or unpredictable. The PMF – CELP coding, which will replace the G.729 pitch modelling, evaluates as well as possible the contribution of both codebooks with a lower computation time and allows to reduce the transmission rate.

The PMF procedure is based on waveform comparisons using the input speech s^k and the residual signal r^k of the LP analysis for the k^{th} subframe. To extract the excitation parameters, the algorithm chooses between a periodic or an aperiodic coding, by computing two sub-band filterings, belonging to the frequency band voice [0, 4000] Hz. Their impulse response are denoted h_{pb} and h_{ph} (figure 3.1).


 Figure 3.1 : Impulse Responses h_{pb} (a) and h_{ph} (b)

The PMF – CELP coding, composed of a periodic and aperiodic coding and of a stochastic codebook search, is detailed by the flowchart in figure 3.2.

3.1. Periodic Coding

In this part of the PMF – CELP coding, a closed-loop pitch analysis is used. The optimal waveform v is selected by maximizing the correlation R_a of the perceptually weighted signal s_w with the adaptive codebook components D :

$$R_a(p) = \sum_{n=0}^{N-1} s_w(n) D(n - p_a) \quad (2)$$

where $p_a = 20, \dots, 4N - 1$ and N is the subframe length equal to 40 samples. The vector D is constituted of 140 samples of past reconstructed excitations and N samples of the residual signal, resulting from the LP analysis, which complete the current subframe excitation necessary for the pitch values p_a inferior to the subframe length.

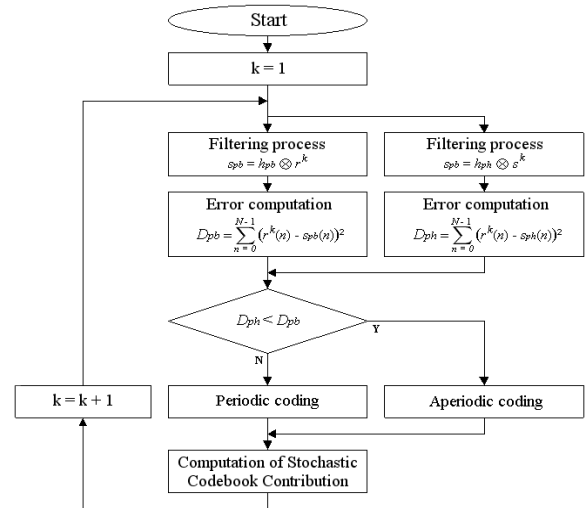


Figure 3.2 : Flowchart of the PMF-CELP coding

3.1.1. PMF-Adaptive Codebook Search

To proceed a more efficient adaptive codebook search, the PMF procedure does not use an open-loop pitch search which limits the closed-loop analysis to a part of the adaptive codebook. However, to optimize the estimation of v , we propose to split the codebook D up into three $2N$ -samples vectors : $d_k(n) = D(n - Nk - 19)$ with $n = 0, \dots, 2N - 1$ and $k = \{1, 2, 3\}$. The correlation R_d will be computed only if the vector d_k contains a part of the past excitation estimated with the periodic coding :

$$R_d(Nk + 19 - p_d) = \sum_{n=0}^{N-1} s_w(n) d_k(n + p_d) \quad (3)$$

where $p_d = 0, \dots, N - 1$. The integer value $p_{d_{opt}}$, which maximizes the correlation R_d , allows to extract the adaptive codebook contribution $v(n) = D(n - Nk - 19 + p_{d_{opt}})$ with $n = 0, \dots, N - 1$.

3.1.2. Fractional Pitch Values

To take into account a pitch period which will be not a multiple of the sampled frequency (p is not an integer), this part presents an adaptive codebook search with fractional pitch values used in the PMF – CELP coding.

To achieve a better pitch estimation, the PMF procedure can introduce higher resolution delays, as the G.729 coder, by determining a fractional pitch of $\frac{1}{3}$ resolution. The adaptive codebook contribution is optimized, with a fractional delay $p_f = \{-\frac{1}{3}, 0, \frac{1}{3}\}$, by interpolating the previous waveform v selected with integer pitch values.

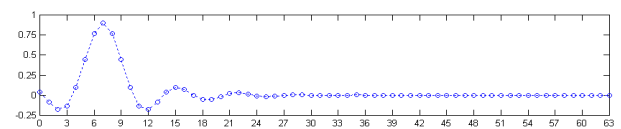


Figure 3.3 : Impulse Response of Interpolation Filter

The vector v is filtered through an interpolation filter, whose the impulse response h_{ip} is based on $\text{sinc}(x)$ function, to determine three waveforms w_j with $j = \{0, 1, 2\}$. The new optimal waveform v is then selected, by minimizing the quadratic error $E_j = \sum_{n=0}^{N-1} (s_w(n) - w_j(n))^2$, and its associated gain factor β is estimated.

3.2. Fixed Codebook Contribution

The fixed codebook contribution is selected with the *G.729* method [3]. The computation of the pulses sequence c will not be detailed because no modifications have been made.

The excitation code vector, composed of non-zero pulses with sign ± 1 , is generated by searching four pulses positions in an algebraic codebook.

$$c(n) = \left(\pm \sum_{j=0}^3 \vartheta_j \delta_{n,m_j} \right) \otimes h_w \quad (4)$$

where $n = 0, \dots, N - 1$, $\delta_{n,m}$ denotes the *Kronecker* delta, ϑ the gains of each pulse, h_w the impulse response of the perceptually weighting filter and the operator \otimes the convolution. After the selection of the optimal waveform c , its gain factor G is estimated.

A slight improvement in the voiced sounds quality may be obtained by processing a filtering of the current excitation. As the adaptive codebook contribution is generally more energetic than the fixed codebook components, a low-pass filter can be introduced to attenuate the higher frequencies and increase the pitch prediction.

3.3. Aperiodic Coding

When the speech signal contains only an unvoiced signal or a background noise, it is preferable to eliminate the adaptive codebook contribution to the excitation. It is more efficient to select a white-noise sequence to excite the synthesis filter. The excitation signal can be determined as :

$$u = \beta (nx \otimes h_w) \quad (5)$$

with x a gaussian white noise sequence and n its appropriate gain. However, the fixed codebook contribution will be conserve to adjust at best the unpredictable excitation frame. The excitation is then constructed as :

$$u = \beta (nx_p \otimes h_w) + Gc \quad (6)$$

where the vector $x_p(n) = x(n) \left(1 - \sum_{j=0}^3 \delta_{n,m_j} \right)$ with $n = 0, \dots, N - 1$ and m_j the four positions determined in the fixed codebook contribution computation.

4. Number Theoretic Transform

To develop the previous *PMF* procedure and the *ITU G.729* codec in fixed-point arithmetic with a low computational complexity, we propose to implement the different algorithms by using Number Theoretic Transforms (*NTT*) which will replace Discrete Fourier Transforms (*DFT*) in the different convolution computations.

4.1. Definitions

A Number Theoretic Transform [2] presents the same form as a *DFT* but is defined over finite rings. All arithmetic must be carried out modulo M , which may be equal to a prime number or to a multiple of primes, since an *NTT* is defined over the Galois Field $GF(M)$. An *NTT* of a discrete time signal x and its inverse are given respectively by :

$$X(k) = \left\langle \sum_{n=0}^{N-1} x(n) \alpha^{nk} \right\rangle_M \quad (7)$$

$$x(n) = \left\langle N^{-1} \sum_{k=0}^{N-1} X(k) \alpha^{-nk} \right\rangle_M \quad (8)$$

where $n, k = 0, \dots, N - 1$.

The *DFT* N^{th} root of the unit in \mathbb{C} , $e^{j\frac{2\pi}{N}}$, is replaced by the N^{th} root of the unit over $GF(M)$ represented by the generating term α , which satisfies the equality $\langle \alpha^N = 1 \rangle_M$ where N is the length of the tranform and $\langle \cdot \rangle_M$ denotes the modulo M operation.

Note that an *NTT* has similar properties as the *DFT* such as the periodicity, symmetry or shift properties. Moreover, an *NTT* admits the Cyclic Convolution Property [6] [7] :

$$U \otimes V = T^{-1} \{ T(U) \bullet T(V) \} \quad (9)$$

where U and V represent both sequences to be convolved, T and T^{-1} are the forward and inverse *NTT* respectively. The operator \bullet denotes the term by term multiplication.

4.2. Fermat Number Transform

The particular modulo equal to a Fermat number, $F_t = 2^{2^t} + 1$ with $t \in \mathbb{N}$, involves the highly composite transform lengths N and the values of α can be equal to a power of 2, hence allowing the replacement of multiplications by bit shifts. The parameters values of a Fermat Number Transform (*FNT*), defined over $GF(F_t)$, are given by $N = 2^{t+1-i}$ and $\alpha = 2^{2^i}$ with $i < t$ (Table I).

Table I : Possible Combinations of *FNT* Parameters

t	modulo F _t	N for α = 2	N for α = 4	N for α = √2
2	2 ⁴ +1=17	8	4	16
3	2 ⁸ +1=257	16	8	32
4	2 ¹⁶ +1	32	16	64
5	2 ³² +1	64	32	128
6	2 ⁶⁴ +1	128	64	256

Note that a *FNT* computation needs about $N \log_2 N$ simple operations (bit shifts, additions) but no multiplication, while a *DFT* requires in order $N \log_2 N$ multiplications.

Moreover, a fast *FNT*-type computational structure (*FFNT*) similar to the Fast Fourier Transform exists. Then, available *FFT VLSI* hardware structure for real-time implementation of the *FFNT* may be adopted. Some tests have shown that a *FFNT*-based convolution reduces the computation time by a factor of 3 to 5 compared to the *FFT* implementation [8].

5. Numerical Results

Numerical simulations have been conducted to evaluate the performances of the proposed *PMF* procedure implemented with *FNT* through the *ITU G.729* coder [3]. In these tests, the *G.729* pitch modelling is compared to the *PMF - CELP* coding into *MatLab* software. Both methods are evaluated, for the same sentence (figure 5.1), by computing objective measures such as the *pitch prediction gain* or the *spectral distortion*.

5.1. Performances Comparisons

In figure 5.2, the plots represent the *pitch prediction gain* G_{LT} obtained with both pitch modelling procedures.

$$G_{LT} = 10 \log_{10} \left(\frac{\sum_{n=0}^{N-1} s_k^2(n)}{\sum_{n=0}^{N-1} (s_k(n) - \hat{s}_k(n))^2} \right) \quad (10)$$

where s_k and \hat{s}_k are the k^{th} subframe of the input and synthesized speech signals respectively.

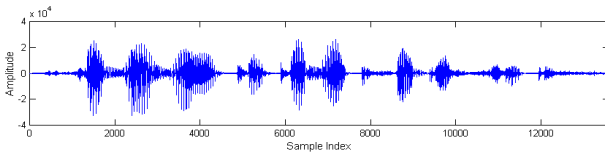


Figure 5.1 : Input speech signal

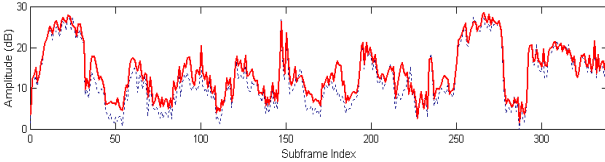


Figure 5.2 : Pitch prediction Gain

The dotted curve represents the standard *G.729* pitch prediction results and the bold plot corresponds to the observed *PMF-CELP* coding performances.

As the new *PMF* structure induces slight improvements in the reconstructed signal (figure 5.2), both pitch modelling give very close listening quality. Subjective speech evaluations have been conducted and show that the *PMF-CELP* coder, with a reduced computation complexity, synthesizes a pleasant speech.

Some tests have been realized with other input sentences, pronounced by males and females, and with signals in presence of noise. All observations confirm the *PMF-CELP* coding is efficient and robust.

5.2. FNT-based PMF Coding Implementation

The *PMF* procedure has been developed to allow its implementation owing to *FNT* and operate with 32-bit numbers. All arithmetic and *FNT* computations will be then carried out modulo F_5 . Various convolutions involved in the *PMF-CELP* coding, can be easily implemented using *FNT*. To facilitate their process computations, the length of three filters impulse responses h_{pb} , h_{ph} and h_{ip} , illustrated in figures 3.1 and 3.3, have been chosen equal to 64, with 16-bits quantized samples.

The figure 5.3 shows the operations numbers required for three pitch modelling methods. The figure give the numbers of multiplications on the left and of basic operations (additions, bit shifts) on the right.

The standard *G.729* pitch analysis is represented by the plots with \square markers. The other plots give the operations numbers for both implementations of the *PMF* procedure. A *FFT*-based realization and a *FNT*-based implementation are drawn with the markers \diamond and \circ respectively.

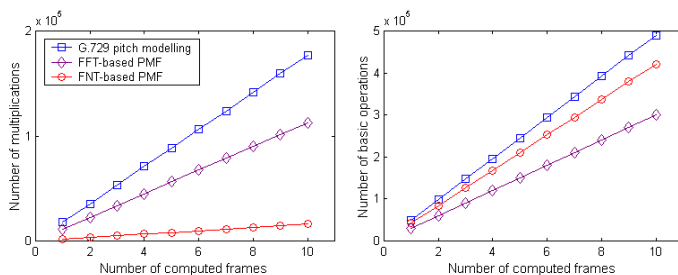


Figure 5.3 : Operations required for the pitch modelling

Although the recent *DSPs* become more and more powerful, the multiplications remain more complicated than basic operations. Hence, to evaluate the different procedures computational cost, the number of multiplications will be particularly considered.

Note that the computation gain is more significant for the filtering and correlation process. Comparing to the *DFT* implementation, the proposed *FNT*-based *PMF-CELP* coding involves a significantly reduction of the multiplications number, by a factor higher than 6 for each frame.

6. Conclusion

In this paper, an efficient implementation of a pitch modelling for a Code-Excited Linear Prediction coder is proposed. A design of a new *Pitch Modelling* by *linear Filtering*, which differentiates the voiced and unvoiced speech frames coding, has been presented. The robust *PMF* procedure reduces the transmission rate and the coding complexity of the speech coder. Following stage would be to decrease the computation time of the fixed codebook search [9].

Objective and recent subjective evaluations of the *PMF-CELP* coding through *CS-ACELP G.729* coder reveal an interesting quality of the reconstructed speech signal. Although the selected application is the *ITU G.729* coder, our *PMF* procedure can be adapted to other *CELP*-type coder.

Moreover, an implementation using Fermat Number Transforms, which reduce the computational cost of the convolution computations, involves the *PMF-CELP* computation complexity is considerably reduced. Note that the *FNT*-based implementation could be beneficial to other functions of speech coders [10].

7. References

- [1] P.Kroon, W.B. Kleijn, "Linear-prediction based analysis-by-synthesis coding", in *Speech Coding and Synthesis*, pp79-115, Elsevier, 1995.
- [2] G.A. Julien, "Number Theoretic Techniques in Digital Signal Processing", *Advances in Electronics and Electron Physics*, Academic Press Inc., vol. 80, Chapter 2, pp. 69-163, 1991.
- [3] ITU-T Recommendation. G.729, "Coding of Speech at 8 kbits/s using Conjugate Algebraic Code-Excited Linear Prediction (CS-ACELP)", 1996.
- [4] R. Salami, C. Laflamme, B. Bessette, J.P. Adoul, "ITU-T G.729 Annex A : Reduced complexity 8 kbits/s CS-ACELP codec for digital simultaneous voice and data", *IEEE Communications Magazine*, vol. 35, pp. 56-63, September 1997
- [5] R.P. Ramachandran, P. Kabal, "Pitch prediction filters in speech coding", *IEEE Trans. ASSP-37*, pp. 467-477, 1989.
- [6] R. Blahut, "Fast algorithms for digital signal processing", Addison-Wesley Publishing Company, 1985.
- [7] W. Shu, Y. Tianren, "Algorithm for linear Convolution using Number Theoretic Transforms", *Electronics Letters*, vol. 24, N°5, March 1988
- [8] R.C. Agarwal, C.S. Burrus, "Fast convolution using Fermat number transform with application to digital filtering", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. ASSP-22, N°2, pp. 87-97, 1974
- [9] Nam Kyu Ha, "A Fast Search Method Of Algebraic Codebook By Reordering Search Sequence", *ICASSP'99*, vol. 1, pp. 21-24, Phoenix, USA.
- [10] G. Madre, E.H. Baghious, S. Azou, G. Burel, "Linear Predictive Speech Coding using Fermat Number Transform", *EURASIP Conf. on Multimedia Communication*, Zagreb, Croatia, 2003.