# AN INTRINSICALLY RELIABLE AND FAST ALGORITHM TO COMPUTE THE LINE SPECTRUM PAIRS (LSP) IN LOW BIT RATE CELP CODING

**A. GOALIC and S. SAOUDI**

Department of Signals and Communications
ENST-Br BP 832 -29285 Brest -BRITTANY, FRANCE
Andre.Goalic@ENST-BRETAGNE.fr

## ABSTRACT

The Code Excited Linear Prediction Coder (CELP) makes it possible to synthesize good quality speech at low bit rates. In such a case, speech quality mainly depends on spectral envelope design accuracy. Different kinds of parameters belonging to the parametrical domain (linear prediction coefficients $\{a_i\}$-LPC - parameters), to the time domain (PARtial CORrelation - PARCOR - parameters) and to the frequency domain (Line Spectrum Pairs - LSP -parameters) are used to design the vocal tract transfer function. The latter present interesting properties both for quantization and interpolation, confirming their increasing utilization in low bit rate speech coding. In real time processing, efficient methods must be used to compute this set of parameters. The purpose of this paper is to show attractive properties in real time processing for Line Spectrum Pair computing (LSP). The possibility of computing each LSP parameter independently characterizes the intrinsic reliability of this method based on the use of the Split Levinson Algorithm. This method is compared to the one using the Chebyshev polynomials.

## 1. LINE SPECTRUM PAIRS

In the speech coding domain, a human vocal tract function transfer can be designed using different sets of parameters. LPC parameters are too dynamic to be used efficiently in speech coding. The bounded property of PARCOR parameters make them very interesting in the coding process; unfortunately they lead to spectral impairment as soon as interpolation is required, for example at very low bit rates. LSPs seem to be well-suited parameters at low and very low bit rate speech coding, where they are now widely used. Belonging to the frequency domain, they require complicated algorithms in order to be obtained. Let us recall the computing process.

The human vocal tract pth-order inverse transfer function can be expressed using LPC parameters $\{a_i\}$:

$$A_p(z)=1+\sum_{i=1}^{p} a_i z^{-i}$$

$A_p(z)$ may be decomposed into a set of two inverse transfer functions, corresponding respectively to complete opening
$[P(z)]$ and complete closing $[P^*(z)]$ of the glottis. The inverse transfer function can be expressed as :

$$A_p(z)=[P(z)+P^*(z)]/2$$

The two inverse transfer functions roots directly give the Line Spectrum Pairs $w_i$ (for an even value of p) :

$$P^*(z)=\left(1-z^{-1}\right)\prod_{i\,even}\left(1-2\cos w_i z^{-1}+z^{-2}\right)$$

$$P(z)=\left(1+z^{-1}\right)\prod_{i\,odd}\left(1-2\cos w_i z^{-1}+z^{-2}\right)$$

with the ordering properties :

$$0<w_1<w_2......<w_p<\pi$$

The LSPs parameter computing process begins with :
- Hamming time filtering window.
- Autocorrelation function computation of the first p+1 values.

According to the method used, the computing process follows different steps :

- the first method(section 2), widely used in coders and vocoders, first computes the Linear Prediction Coefficients using the Levinson Durbin algorithm, and secondly, after developing P(z) and $P^*(z)$ in cosine functions, uses CHEBYSHEV polynomials to make possible real-time processing. Finally a dichotomy method is used to compute P(z) and $P^*(z)$ roots - KABAL et al. method - [1].

- the second (section 3), recently introduced, uses the Split Levinson algorithm to compute only one inverse transfer function coefficient. The recurrence relation between different transfer function orders makes it possible to design two tridiagonal matrices, each eigenvalue of which depends only on an LSP function. Matrice elements property allows the utilization of the bisection method to compute rapidly the LSPs - SAOUDI et al. method - [2].

With the same initial conditions and computation accuracy, the two methods give the same results. The $10^{th}$ prediction order is usually used in low bit rate speech coding (p=10).

0-7803-2431-5/95 $4.00 © 1995 IEEE

Finally, before concluding, section 4 compares the two methods and give results achieved in real time implementation.

## 2. KABAL et al. method

This is probably the most widely used method of computing LSP parameters. It includes the following steps:

*2.1 LPC parameter Computation*

Linear Prediction Coefficients $\{a_i\}$ are computed using the Levinson Durbin algorithm.

*2.2 The development of a transfer function cosine*

Expressed as a LPC parameter function, each transfer function is as follows :

$$P(z) = a_0(z^0 + z^{-(p+1)}) + \sum_{i=1}^{p}(a_i + a_{p+1-i})z^{-(p+1)}$$

$$P^*(z) = a_0(z^0 - z^{-(p+1)}) + \sum_{i=1}^{p}(a_i - a_{p+1-i})z^{-(p+1)}$$

One root of each transfer function is known, so they can be expressed as :

$$P(z) = (1+z^{-1})\sum_{k=0}^{p} l_k z^{-k} = (1+z^{-1})Q(z)$$

$$P^*(z) = (1-z^{-1})\sum_{k=0}^{p} l_k^* z^{-k} = (1-z^{-1})Q^*(z)$$

Relations between parameters $l_k$, $l_k^*$ and the property of the located roots on the unit circle leads to $Q(z)$ and $Q^*(z)$ cosine development as follows :

$$Q(w) = e^{-j\frac{pw}{2}}\left(\sum_{k=0}^{\frac{p}{2}-1} 2l_k \cos(\tfrac{p}{2}-k)w + l_{\frac{p}{2}}\right) = Q(z)\big|_{z=e^{jw}}$$

$$Q^*(w) = e^{-j\frac{pw}{2}}\left(\sum_{k=0}^{\frac{p}{2}-1} 2l_k^* \cos(\tfrac{p}{2}-k)w + l_{\frac{p}{2}}^*\right) = Q^*(z)\big|_{z=e^{jw}}$$

Finally, $R(w)$ and $R^*(w)$ polynomial roots give the LSPs :

$$R(w) = \sum_{k=0}^{\frac{p}{2}-1} 2l_k \cos(\tfrac{p}{2}-k)w + l_{\frac{p}{2}}$$

$$R^*(w) = \sum_{k=0}^{\frac{p}{2}-1} 2l_k^* \cos(\tfrac{p}{2}-k)w + l_{\frac{p}{2}}^*$$

Requiring trigonometric function evaluation, this method seems to be difficult and time consuming in real time processing. To solve this problem, Kabal and Ramachandran use the Chebyshev polynomials.

*2.3 Chebyshev Polynomials*

Chebyshev polynomials satisfy the following recursion order :

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$$

where $T_k(x) = \cos(kw)$ is a $k^{th}$ -order Chebyshev polynomial in x ($x = \cos(w)$). The two previous polynomials can now be expressed in terms of Chebyshev polynomials :

$$R(w) = \sum_{k=0}^{\frac{p}{2}} c_k T_k(x)$$

$$R^*(w) = \sum_{k=0}^{\frac{p}{2}} c_k^* T_k(x)$$

$c_k$, $c_k^*$ are expressed using LPC parameters $\{a_i\}$. The following recurrence relationship makes it possible to compute polynomials without evaluating the cosine function :

$$b_k(x) = 2xb_{k+1}(x) - b_{k+2}(x) + c_k$$

with $b_6(x) = b_7(x) = 0$. This recursion is used to calculate $b_0(x)$ and $b_2(x)$ before computing the two following polynomials :

$$R(w) = \sum_{k=0}^{\frac{p}{2}}\left(b_k(x) - 2xb_{k+1}(x) + b_{k+2}(x)\right)T_k(x) = \frac{b_0(x) - b_2(x) + c_0}{2}$$

$$R^*(w) = \sum_{k=0}^{\frac{p}{2}}\left(b_k^*(x) - 2xb_{k+1}^*(x) + b_{k+2}^*(x)\right)T_k(x) = \frac{b_0^*(x) - b_2^*(x) + c_0^*}{2}$$

### *2.4 Line Spectrum Pair computation*

The last steps of this method include $R(w)$ and $R^*(w)$ root evaluation. To understand the computing process well, let us recall that $R(w)$ and $R^*(w)$ roots interlace on the unit circle. The dichotomy method used to compute the LSPs has to take this fact into account. As roots lie in the ]-1,+1] range, the dichotomy method is applied between these two limits. The process begins to evaluate $R(w)$ the first root of the polynomial and the first root search for $R^*(w)$ polynomial starts from the position of the $R(w)$ root just found. The process continues as before but interchanges polynomial roles as each root is found.
To be sucessfully performed and not to miss any roots, the dichotomy method has to take into account :

- the minimal distance between two roots of the same polynomial.
- the minimal distance between pairs of roots (one belonging to $R(w)$, the other to $R^*(w)$). This is to avoid missing a root when switching the search from one polynomial to the other. Let the roots be denoted by $\{r_i\}$ and ordered $r_i > r_{i-1}$. According to their parity they belong either to $R(w)$ or $R^*(w)$.

To make possible real time processing on existing processors two iterative steps $(\Delta, \delta)$ must be taken :

- the first must be smaller than the minimal distance between two roots of the same polynomial $(\Delta < (\min(r_{i+2} - r_i))$ ; $i = 0,1)$.
- the second must be smaller than the minimum spacing between pairs of roots $(\delta < (\min(r_{i+1} - r_i))$ ; $i = 0,1)$. Both $\Delta$ and $\delta$ depend on the properties of the speech database statistics.

## 3. SAOUDI et al. method

The classical Levinson Durbin algorithm is known to be redundant in complexity[3]. It can be split into two parts (symmetric and antisymmetric forms), and only one of them has to be computed to evaluate the human vocal tract inverse transfer function. The two inverse transfer functions previously defined, can be expressed as :

$$P(z) = \sum_{i=0}^{P+1} p_{p+1,i} z^{-i}$$

$$P^*(z) = \sum_{i=0}^{P+1} p^*_{p+1,i} z^{-i}$$

Both are computed recursively using the Split Levinson algorithm symmetric and antisymmetric form. The two algorithms also compute the reflection coefficients if required.

### *3.1 Recurrence relations*

The two forms make it possible to establish a three term recurrence relation between different orders of the inverse transfer function of course for each one :

$$P_{k+1}(z) - (1 + z^{-1})P_k(z) + \alpha_k P_{k-1}(z) = 0$$

$$P^*_{k+1}(z) - (1 + z^{-1})P^*_k(z) + \alpha^*_k P^*_{k-1}(z) = 0$$

The two set of inner parameters are evaluated while computing inverse transfer functions. Using a single change of variable, two real polynomials linked by a three-term recurrence relation, can be designated :

$$H_{k+2}(x) - (x - \alpha_k - \alpha_{k+1} + 2)H_k(x) + \alpha_{k-1}\alpha_k H_{k-2}(x) = 0$$

$$H^*_{k+2}(x) - (x - \alpha^*_k - \alpha^*_{k+1} + 2)H^*_k(x) + \alpha^*_{k-1}\alpha^*_k H^*_{k-2}(x) = 0$$

These relations lead to two tridiagonal matrices, each eigenvalue of which depends only on an LSP function :

$$\begin{bmatrix} 2\alpha_1+\alpha_2-2 & 1 & 0 & & \\ \alpha_2\alpha_3 & \alpha_3+\alpha_4-2 & 1 & & \\ 0 & & & & \\ & & & & \\ & & 0 & \alpha_{p-2}\alpha_{p-1} & \alpha_{p-1}+\alpha_p-2 \end{bmatrix}$$

$$\begin{bmatrix} \alpha^*_2-2 & 1 & 0 & & \\ \alpha^*_2\alpha^*_3 & \alpha^*_3+\alpha^*_4-2 & 1 & & \\ 0 & & & & \\ & & & & \\ & & 0 & \alpha^*_{p-2}\alpha^*_{p-1} & \alpha^*_{p-1}+\alpha^*_p-2 \end{bmatrix}$$

### *3.2 LSP computation*

The eigenvalues of the two matrices only depend on the values of the set of parameters $\alpha$ and $\alpha^*$. Each Split Levinson algorithm form can compute reflection coefficients, creating an easy link between the two sets of parameters. Thus only one Split Levinson algorithm form makes it to compute possible $\alpha$ and $\alpha^*$. Therefore the antisymmetric Split Levinson algorithm form (the symmetric form includes a larger number of instructions) is the fastest way to compute the two sets of parameters leading to the Line Spectrum Pairs.

### *3.3 Bisection method*

Once the problem has been posed in matricial form, fast and powerful methods can be used to compute eigenvalues. Responding to utilization conditions, the fastest eigenvalue computation method - the bisection method - is successfully used here to compute the Line Spectrum Pairs. Indeed matrix coefficients $\alpha$ and $\alpha^*$ respond to the following relation :

$$0 < \alpha, \alpha^* < 4$$

Let us recall the main advantages of the bisection method for eigenvalue computation, which makes it possible :

- to know the number of eigenvalues included in an interval, and using an iterative processing to isolate the $i^{th}$ value independently of the others.
- to compute the $i^{th}$ eigenvalue with the required accuracy in this interval.

## 4. COMPARISON WITH REAL TIME PROCESSING

### *4.1 Intrinsic reliability*

The main advantage of the SAOUDI et al. method is its intrinsic reliability. Due to use of the matrix formulation and bisection method, this method evaluate each LSP parameters independently without taking into account any distances between consecutive LSPs. On the contrary KABAL et al. method must take into account these distances not to miss any roots. To avoid possible divergences, tests even if time consuming, must be implemented.

### *4.2 Signal Statistics Independence*

This features comes directly from the first point, in order not to miss roots. As far as the distance between consecutive roots has to be taken into account, the method remains linked with the speech database. This is the case for the KABAL et al. method.

### *4.3 Computation speed*

Table 1 shows the computational requirements measured on the test bench. Both coder and decoder use only a single fixed point Digital Signal Processor MOTOROLA 56001 (clock frequency : 27 MHz) [4]. The number in brackets represents whole coder computational requirements in terms of percentage. To avoid divergences, the dichotomy method often has to compute the polynomials though computational time is drastically reduced by using two incremental steps. Accuracy in LSP determination varies from 5 Hz in the middle frequencies to 20 Hz at low and high frequencies. Our process only computes the Line Spectrum Pair cosine function (CLSP) because the decoder only uses the CLSP.

### *4.4 Digital acoustic underwater phone*

The new CLSP computing process is included in a new digital acoustic underwater phone we are designing at our institute. The coder bit rate is presently 5.45 Kbit/s and the whole system including QPSK modulation has been tested successfully both in a testpool and in BREST bay using acoustic waves.

| Algorithm | LPC analysis M.I.P.S. | CLSP extraction only (M.I.P.S.) |
|---|---|---|
| SAOUDI et al. | 1.43 (10.6%) | 1.1 (8.1%) |
| KABAL et al. | 2.1 (15.6%) | 1.77 (13.1%) |

Table 1 : Computational requirements

## 5. CONCLUSION

This paper has compared two algorithms for computing the Line Spectrum Pairs parameters in a real time environment. Without increasing the complexity, the new one based on the Split Levinson algorithm and the bisection method shows interesting properties and appears to be well-suited for real time applications : intrinsic reliability, independence from the signal statistics and fast computation.

## ACKNOWLEDGEMENTS

The authors thank Janet OMROD for her kind assistance with the english of this paper.

## MAIN REFERENCES

[1] P. Kabal and R.P. Ramachandran,"The computation of line Spectral Frequencies using Chebyshev polynomials", Vol. ASSP-34, N° 6, pp. 1419-1425, December 1986.

[2] S. Saoudi, J.M. Boucher and A. Le Guyader,"A new efficient algorithm to compute the LSP parameters for speech coding", Signal Processing, Vol. 28, No 2, pp.201-212, August 1992.

[3] P. Delsarte and Y. Genin,"The split Levinson algorithm", Proc. ICASSP'86, June 1986, pp. 470-478.

[4] A. Goalic "Etude et réalisation de Codeur/décodeur de parole à bas débit pour la téléphonie acoustique sous-marine". Thèse de doctorat, Université de Bretagne Occidentale BREST, Juin 1994 (Study and development of a low bit rate speech coder/decoder for an acoustic underwater phone)(Doctoral thesis...).