An Algorithm for the Computation of LSP parameters

Khoudir Benbellil

الملخّـص

نقترح في هذا المقال خوارزمية لحساب عوامل LSP باستعمال الحساب العقدي كما نناقش مسألة حساب جذور LPC.

الكلمات المفاتيح : الحساب العقدي، عوامل LPC، عوامل LSP.

An Algorithm for the Computation of LSP parameters

Khoudir BENBELLIL

Centre de Recherche Scientifique et Technique pour le Développement de la Langue Arabe, Alger

benbellilk@yahoo.fr

Abstract

In this paper, we propose an FFT based algorithm for the computation of LSP parameters using complex analysis tools. The computation of LPC roots are also discussed.

Keywords: LPC, LSP, LSF, Cauchy formulae, homomorphic analysis, speech parameterizations.

1. Introduction

The purpose of this paper is to present an efficient algorithm using complex analysis tools for the computation of roots of polynomials [2] and its application in the LSP and LPC analysis framework. LPC and LSP (or LSF) parameters are compact representations of speech signals used in synthesis, coding and recognition of speech. LSP and LSF parameters enjoy good properties of distortion and interpolation. The relation between LPC coefficients and LSP coefficients are given by[th]:

$$P(z) = A(z) + z^{-(p+1)}A\left(\frac{1}{z}\right) \qquad (1)$$

$$Q(z) = A(z) - z^{-(p+1)}A\left(\frac{1}{z}\right) \qquad (2)$$

$$A(z) = \frac{1}{2}(P(z) + Q(z)).$$

$P(z)$ and $Q(z)$ are called palindromic polynomials. In formant computing, a formant occuring at a root of LPC polynomial [1], [2] there is a pair of zeros (for P and Q) near this root.

Methods for finding roots of polynomials and other transcendental functions are numerous [3], [6], [8]. Since our work deals only with polynomials defined on the complex plane, it is natural to use contour integration (Cauchy's theorem) to compute roots and to establish important estimations also [4], [7]. Moreover, by an appropriate choice of the contour integration like the circle, the use of contour integration method reduce to FFT transform computation. In the sequel, we present first the idea of the method for then we particularize it for the root finding task of LSP and LPC polynomials. In fact, the only necessary tool in the complex analysis framework is the well known Cauchy formulae given by:

$$f^{s}(\zeta) = \frac{1}{2\pi i}\oint_C \frac{f(z)}{(z-\zeta)^{s+1}}dz \qquad (3)$$

Here, the integration is taken over the contour C bordering (Figure 1) an open domain in the complex plane. The orientation of integration, s is an integer and the s-order derivative is given by:

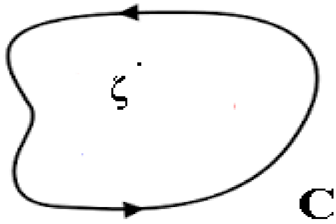

Figure 1 : Cauchy's theorem

$f^{(s)}(\zeta) = \frac{d^s}{dz^s} f(z)\Big|_{z=\zeta}$ and $f^{(0)}(\zeta) = f(\zeta)$

z is a complex variable running over C in counter-clockwise sense and ζ is a fixed point inside the domain. From these formulae we can deduce useful identities as:

$\frac{1}{2\pi i} \oint_C \frac{1}{(z-\zeta)^s} dz = 0$ for all $s \neq 1$ and

$$\frac{1}{2\pi i} \oint_C \frac{1}{z-\zeta} dz = 1.$$

In this domain, P has d roots which are distinct or not, by virtue of the fundamental theorem of Algebra, hence:

$$P(z) = \prod_j (z - z_j)^{r_j}$$

$$Log(P(z)) = \sum_j r_j \, Log(z - z_j) + 2k\pi i$$

Adding the factor $2k\pi i$ is important because the logarithm function is a multivalent function in the complex plane. However, this factor disappears in the derivation operation which gets:

$$\frac{d}{dz}(Log(P(z)) = \frac{P'(z)}{P(z)} = \sum_j \frac{r_j}{z - z_j}$$

Hence:

$$N = \frac{1}{2\pi i} \oint_C \frac{P'(z)}{P(z)} dz$$

$$= \sum_j \oint_C \frac{r_j}{z-z_j} dz = \sum_j r_j. \quad (4)$$

We have also:

$$S = \oint_C \frac{zP'(z)}{P(z)} dz =$$

$$\sum_j \oint_C \frac{r_j z}{z-z_j} dz = \sum_j r_j z_j \quad (5)$$

N and S being respectively the number and the sum of the roots of P. The formulas (4) and (5) can be easily evaluated over the circular contour C with radius r (Figure 2) by doing the change of variable $z = re^{i\theta}$. We obtain:

$$N = \frac{1}{2\pi} \int_0^{2\pi} r \frac{P'(re^{i\alpha})}{P(re^{i\alpha})} e^{i\alpha} d\alpha \quad (6)$$

$$S = \frac{1}{2\pi} \int_0^{2\pi} r^2 \frac{P'(re^{i\alpha})}{P(re^{i\alpha})} e^{2i\alpha} d\alpha \quad (7)$$

With these formulas, an algorithm can be devised to find the root of an arbitrary polynomial. For this purpose, we need to realize the same number d of sums of roots as the number of roots. More precisely, let the domain be the disk $|z| \leq r$ (Figure 2). By decreasing the radius of this disk at each step q to a value $r(q)$, the number of roots becomes N_q and their sum becomes S_q giving thus:

$$z_1 + z_2 + \ldots + z_{N_q} = S_q, \; q=1,2,\ldots d. \quad (8)$$

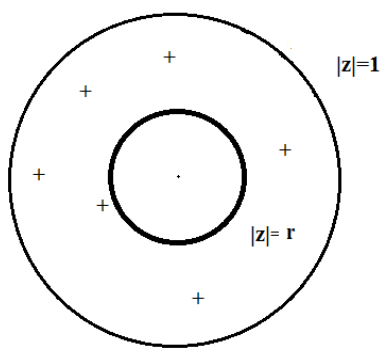

Figure 2 : Roots of polynomial inside a disk

If the decrease of *r(q)* is such that N_q decreases by 1 at each step, we obtain a linear triangular system with *d* equations and *d* unknowns which can be easily resolved giving thus:

$$z_r = S_r - (z_1 + z_2 + \ldots + z_{r-1})$$
$$r=1,2,\ldots d. \qquad (9)$$

For the case of LPC polynomial, the variation of the number of roots at each step is set to 2 because the prediction polynomial has conjugate pairs of zeros since its coefficients are real. The chosen domain is the unit disk $|z| \leq 1$ assuming that the system is a minimal phase system [rab],[ch].

The paper is composed of 3 sections. In the second section we adapt the method of root finding to LSP parameters for speech signals and we establish some useful relations for our algorithm. In the third section we propose a general algorithm for the computation of LSP parameters. The paper ends by a conclusion and some remarks about the application of the algorithm to the LPC case.

2. Roots of palindromic polynomials

In this section, we adapt our approach to the palindromic polynomials (1) and (2). Since these polynomials enjoy particular properties as having all their roots on the circle $|z| = 1$, the contour in Figure 2 is not appropriate. Instead we consider a travelling small circular contour as depicted in Figure 3. The small circle has its center located on the big circle. Moreover, since the coefficients of *P* and *Q* are all real, their roots are conjugate pairs so we need to compute only the roots having an argument between 0 and π on the big circle, the others are computed by symmetry.

The equation of the big circle is $|z| = 1$ while the equation of the small circle is $|z - z_0| = re^{i\alpha}$ where *r* is given by:

$$r = 2\sin((\alpha_2 - \alpha_1)/4) \qquad (10)$$

Hence on the small circle we have:

$$z = z_0 + re^{i\alpha} \qquad (11)$$

where $z_0 = \exp\left(\frac{i(\alpha_1+\alpha_2)}{2}\right)$ is the complex affix of the center of the small circle. Putting these expressions in (4) and (5) we obtain the formulas of the number and the sum of the roots:

$$N = \frac{r}{2\pi}\int_0^{2\pi} G(r,\alpha)\, e^{i\alpha} d\alpha \qquad (12)$$

$$S = \frac{r}{2\pi}\int_0^{2\pi}(z_0 + re^{i\alpha})G(r,\alpha)\, e^{i\alpha} d\alpha \qquad (13)$$

where $G(r,\alpha) = \dfrac{P'(z_0 + re^{i\alpha})}{P(z_0 + re^{i\alpha})} \qquad (14)$

In the next section, we realize the discretization of these formulae and we present an algorithm to find all the roots of the polynomials *P* and *Q* which give directly the coefficients LSF.

3. The general Algorithm of roots' calculation

To evaluate the integrals (12) and (13) by a computer, we replace them by Riemann sums where the variable α runs over the interval $[0,2\pi]$. These Riemann's sums and the expression of the form $P(z_0 + re^{i\alpha})$ can then be evaluated using FFT and/or IFFT algorithms. By a discrete N-points scheme on the small circle given by:

$$\alpha_k = k\frac{2\pi}{N},\ 0 \leq k \leq N-1 \qquad (15)$$

and the approximate differential:

$d\alpha \approx \Delta\alpha = \frac{2\pi}{N}$ we replace N and S in (12) and (13) by their approximants:

$$N_k = \frac{r}{N}\sum_{k=0}^{k=N-1} G(r,k)e^{\frac{ik2\pi}{N}} \quad (16)$$

$$S_k = \frac{r}{N}\sum_{k=0}^{k=N-1} H(r,k)e^{\frac{ik2\pi}{N}} \quad (17)$$

where

$$G(r,k) = \frac{P'(z_{1,2}+re^{\frac{ik2\pi}{N}})}{P(z_{1,2}+re^{\frac{ik2\pi}{N}})} \quad (18)$$

$$H(r,k) = \left(z_{1,2} + re^{\frac{ik2\pi}{N}}\right)G(r,k) \quad (19)$$

$$z_{1,2} = exp\frac{i\pi}{N}(k_1 + k_2) \quad (20)$$

k_1 and k_2 being the points corresponding to the bounds of the sector θ_1 and θ_2 in the scheme (15). Note that the expressions (18) and (19) can easily be evaluated using FFT transforms using the N-scheme (15) since for a polynomial having the form:

$$\sum_{l=0}^{l=d} a_l z^l$$

the values for the scheme (15) are:

$$\sum_{l=0}^{l=d} a_l r^l e^{l\frac{ik2\pi}{N}} = \hat{A}(k)$$

with $0 \le k \le N-1$. Note that $A(k)$ is N times the inverse transform of the vector:

$$A(l) = \begin{cases} a_l r^l \; for \; 0 \le l \le d \\ 0 \; for \; l > d \end{cases}$$

Similarly, the discrete sums (16) and (17) can be evaluated using inverse Fourier transform. For (16) we have:

$$\hat{G}(l) = \frac{1}{N}\sum_{k=0}^{k=N-1} G(r,k)e^{\frac{ik2\pi l}{N}} = \text{IFFT(G)}$$

with $0 \le l \le N-1$ and we deduce the value of N_k:

$$N_k = \frac{r}{N}\sum_{k=0}^{k=N-1} G(r,k)e^{\frac{ik2\pi}{N}}$$
$$= r\hat{G}(1)$$

The sum (17) is evaluated similarly.

Computations carried out in the MATLAB environment showed these formulae give correct results. These results were more accurate in the case of the contour of Figure 3 than in the case of the contour of Figure 2, the reason seems to be the smallness of the circumference of the small circle leading thus to a better precision. To implement an algorithm for computing all the roots we are faced to some difficulties in the general (or LPC) case:

We do not know a priori on which circle a root is localized and this can lead to problems in the contour integration if this contour intercepts a root;

At each step, we must evaluate the number of roots and if this number is one in the general case or two in the conjugate roots pairs case, and then we evaluate their sum. Clearly, this approach needs a big amount of calculations.

The case of LSP parameters is more interesting. Contrary to LPC polynomial,

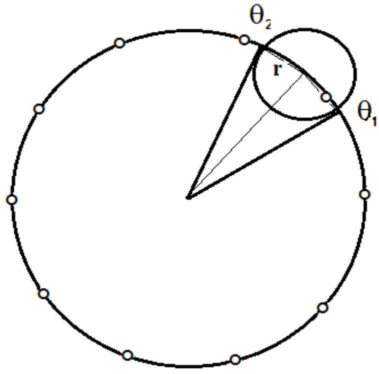

Figure 3 : Computing roots of palindromic polynomials

the palindromic polynomials have all their roots on the circle $|z| = 1$ and between two roots of P there is one root of Q. Here, we propose a general algorithm for finding all the roots for both P and Q:

- We try to divide the arc (0,π) of the big circle into [*d/2*] (integer part of *d/2*) sectors, *d* being the degree of the polynomials i.e. the number of the roots;
- Evaluate the sum of roots in each sector. If the sum is zero, then the sector is void of roots so we remove this sector and we restart with a smaller arc. If the sector contains one root, then the value of the sum is the value of the root and we repeat the search process with the complementary arc with the value of *d* decreased by 1. If the sector contains more than one root, we split the sector into equal parts aiming to isolate one root in one sector;
- When all the roots corresponding to the upper half circle are found, the other roots are found by axial symmetry (the axis is the real line);
- When the roots of P are found, it is easy to find the roots of Q, since between a pair of roots of P there is a unique root of Q.

The localization of the roots of P and Q on the big circle permit us to reduce the amount of computations because only the sum formula is needed. Suppose that a sector (θ_1, θ_2) contains m roots having a sum S, then we have:

$$S = z_1 + z_1 \cdots + z_m$$

and suppose, without restricting generality that the sector is inside the first quadrant (0,π/2).

$$S = \cos(\alpha_1) + \cos(\alpha_2) + \cdots \cos(\alpha_m) + i(\sin(\alpha_1) + \sin(\alpha_2) + \cdots \sin(\alpha_m))$$

The roots can be ordered with respect to their arguments so we have:

$$\sin(\theta_1) \leq \sin(\alpha_1) \leq \sin(\theta_2)$$
$$\sin(\theta_1) \leq \sin(\alpha_2) \leq \sin(\theta_2)$$
$$\vdots$$
$$\sin(\theta_1) \leq \sin(\alpha_m) \leq \sin(\theta_2)$$

Summing these inequalities we find:

$$\mathrm{m}\sin(\theta_1) \leq \mathrm{Im(S)} \leq \mathrm{m}\sin(\theta_2)$$

then $\frac{Im(S)}{\sin(\theta_1)} \geq m$ and $\frac{Im(S)}{\sin(\theta_2)} \leq m$ where $\mathrm{Im(S)}$ denotes the imaginary part of S.

The number of the roots inside a given a sector must verify these two inequalities which permit the determination of this number avoiding thus computing the number of roots.

4. Conclusions

We have presented in this paper an algorithm for computing roots of palindromic polynomials permitting thus the calculation of LSF parameters for speech signals. This approach uses tools of complex analysis and involves FFT algorithm by a suitable choice of the contour integration. We have discussed the general case which is suitable for LPC roots computation and noticed that some difficulties may appear if a root intercepts the contour integration. The situation is more stable for the case of palindromic polynomials where the roots are all on the unit circle. Instead of dividing the disk in several annulus, its is divided in several sectors. Care must be taken however in the case where one or both limits of a sector are roots of these polynomials. In this paper, we have only showed the basic ideas of the method which earns a full study by inspecting other aspects like the stability and the computational complexity.

5. References

[1] Rabiner L. R., Schaffer R. W., Digital Processing of Speech Signals, Prentice-Hall signal processing series, 1978.

[2] Huang X., Acero A., Hon H., “Spoken Language Processing, A Guide To Theory, Algorithms and System Development”, Prentice Hall PTR, Prentice-Hall, Inc. Upper Sadle River, New Jersey 07458.2001, 2001.

[3] http://en.wikipedia.org/wiki/Splitting_circle_method.

[4] Spiegel M. R., Lipschutz S., Schiller J. J., Spellman D., “Complex variables”, Schaum’s outlines serie Mc GRaw Hill, second edition 2009.

[5] Islam T., “Interpolation of linear prediction coefficients for speech coding”,Mc Gill University, Canada, 2000.

[6] Kabal P., Ramachandran R. P., “The Computation of Line Spectral Frequencies Using Chebyshev Polynomials”, Acoustics, Speech and Signal Processing, IEEE Transactions on, vol.34, no.6, pp.1419,1426, Dec 1986.

[7] Snell, R.C.; Milinazzo, F., "Formant location from LPC analysis data," *Speech* and Audio Processing, IEEE Transactions on , vol.1, no.2, pp.129,134, Apr 1993.

[8] Gourdin A., Boumahrat M., “Méthodes numériques appliqués”,Editeur : Tec et Doc, 2emem édition 1999.