Formant Location From LPC Analysis Data

Roy C. Snell, *Member, IEEE*, and Fausto Milinazzo

***Abstract*— The estimation of formant frequencies and bandwidths from the filter coefficients obtained through LPC analysis of speech is discussed from several viewpoints. A new method for locating roots within the unit circle is derived. This algorithm is particularly well suited to computations carried out in fixed point arithmetic using specialized signal processing hardware.**

I. Introduction

IN SPEECH PROCESSING, formants are resonances of the vocal tract. The estimation of their locations and bandwidths (particularly during the production of voiced speech) is important in many applications (see [1], [2], and [4]). The particular application that motivated much of the development presented in this correspondence was a project in the area of speaker verification [13] in which the number of formants present in several selected frequency ranges was used to identify several phonetic events. This application required an algorithm capable of operating in near real time.

A frequently used technique for formant location involves the determination of resonance peaks from the filter coefficients obtained through LPC analysis of segments of the speech waveform [7]. Once the prediction polynomial $A(z)$ has been calculated, the formant parameters are determined either by "peak-picking" on the filter response curve or by solving for the roots of the equation $A(z) = 0$. Each pair of complex roots is used to calculate the corresponding formant frequency and bandwidth.

The computations involved in "peak-picking" consist of either the use of the fast Fourier transform with a sufficiently large number of points to provide the prescribed accuracy in formant locations or the evaluation of the complex function $A(e^{j\theta})$ at an equivalently large number of points [7]. Both computations can be carried out efficiently on a general purpose computer or by a digital signal processing (DSP) chip such as the Texas Instruments TMS320. However, both methods are frequently unable to distinguish closely spaced resonances. A thorough discussion of this problem is given in [8] and [10].

In spite of their advantages, the relative complexity of polynomial root finding techniques frequently precludes them as an approach to formant estimation. The LPC analysis, which can be carried out efficiently, typically produces a predictor polynomial of degree at least 10 which, due to stability requirements, has all its roots within the unit circle. The software used to determine the roots is often a standard general-purpose root solver available in a scientific subroutine package. The computation time for these routines is, in many cases, too long to allow the calculation of resonance peaks in what may frequently involve several hundred LPC analysis frames derived from just a few seconds of speech. In addition, the major drawback to virtually all root-finding algorithms is the requirement that they be implemented in relatively high accuracy floating point arithmetic, making them unsuitable for high speed implementation in the fixed point arithmetic provided on most standard DSP chips.

Manuscript received January 15, 1990; revised February 29, 1992. This associate editor coordinating the review of this paper and approving it for publication was Dr. Mark A. Clements

The authors are with the Mathematics Department, Royal Roads Military College, Victoria, B.C., Canada V0S 1B0.

IEEE Log Number 9206390.

The aim of this article is to present an algorithm for locating formants and estimating their bandwidths. Unlike most root finding methods, this algorithm has minimal precision requirements and need not be implemented in high precision floating point arithmetic and indeed is very well suited for implementation in the fixed point arithmetic available on most DSP chips. The algorithm is described in Section II and test results, indicating the efficiency and accuracy of the method in both a high level implementation and an implementation using fixed point arithmetic on a digital signal processing (DSP) chip, are presented in Section III.

II. Root Location by Contour Integration

A. Relationships of Formant Frequencies/Bandwidths to Root Locations

A number of standard references on speech processing (see, for example, [7] and [12]) provide the following transformation from complex root pairs $z = r_0 e^{\pm\theta_0}$ and sampling frequency f_s to formant frequency F and 3-dB bandwidth B:

$$F = \frac{f_s}{2\pi}\theta_0 \,\text{Hz} \tag{1}$$

$$B = -\frac{f_s}{\pi}\ln r_0 \,\text{Hz}. \tag{2}$$

Assuming that the prediction polynomial corresponds to a second-order all-pole system, it may be shown that the formant frequency and 3-dB bandwidth are

$$\hat{F} = \frac{f_s}{2\pi}\arccos\left[\cos\theta_0 \frac{(r_0^2+1)}{2r_0}\right] \tag{3}$$

$$\hat{B} = \frac{f_s}{2\pi}|\arccos(x_1) - \arccos(x_2)| \tag{4}$$

where

$$x_1 = \frac{(1+r_0^2)}{2r_0}\cos\theta_0 + \frac{(1-r_0^2)}{2r_0}\sin\theta_0$$

and

$$x_2 = \frac{(1+r_0^2)}{2r_0}\cos\theta_0 - \frac{(1-r_0^2)}{2r_0}\sin\theta_0.$$

It is easy to see that (3) and (1), and (4) and (2) give the same result in the limit as $r_0 \to 1$. How well each pair of equations agree for roots located close to the unit circle ($r_0 \approx 1$) may be seen by setting $r_0 = 1 - k$ and comparing the corresponding Taylor series expansions of these equations about $k = 0$. A short calculation yields the approximations

$$|\hat{F} - F| = \frac{f_s}{4\pi} k^2 |\cot\theta_0| |1 + k + O(k^2)| \tag{5}$$

and

$$|\hat{B} - B| = \frac{f_s}{3\pi} k^3 (1 + 3\cot^2\theta_0) |1 + O(k)|. \tag{6}$$

The estimates given by (5) and (6) indicate good agreement between the two methods of computing the formants and the bandwidths when the corresponding root is near the unit circle ($r_0 \approx 1, k \approx 0$). For fixed r_0, the differences are a minimum for $\theta_0 = \pi/2$ (corresponding to half the Nyquist frequency). As is to be expected the estimates break down at 0 and π (formant frequencies near 0 or the Nyquist frequency).

An examination of the root locations derived from 3400 frames of LPC analysis data showed that 76% of the roots were of magnitude greater than 0.85; 64% greater than 0.90; and 41% greater than 0.95. Thus, for data derived from the LPC analysis of speech, the errors incurred in determining formant frequencies and bandwidths by using (1) and (2) rather than (3) and (4) are not significant.

We emphasize that (3) and (4) correspond to the formant frequency and bandwidth for a single second-order factor of the prediction polynomial $A(z)$, but that interactions among adjacent roots produce an overall response in which resonance peaks are shifted and 3-dB bandwidth values undergo major changes. For this reason, it is unrealistic to demand too high a degree of accuracy in the determination of formant frequencies and their corresponding bandwidths.

B. Algorithm Overview

The method described in this section permits the direct computation of the number of roots of a polynomial $P(z) = 0$ that lie within any closed curve. We shall concentrate on the case where the polynomial $P(z)$ is the predictor polynomial obtained from LPC analysis of a speech waveform. In one application we will indicate how the method can be used to determine the exact number of formants in a given frequency range and, in a second, we extend the method to calculate the locations of all formants determined by the LPC analysis.

In both applications, the closed curves of interest are the boundaries of sectors of the unit circle consisting of rays at angles θ_1 and θ_2, with $0 < \theta_1 < \theta_2 < \pi$, and the arc of the circle. Since formants correspond to conjugate pairs of complex roots, only sectors in the upper half of the unit circle need to be considered.

If the information of interest is the number of formants in the selected frequency range, then the algorithm that we shall describe provides the answer directly. However, the information most frequently required is a list of all formant frequencies for the given analysis frame. In this case, the technique is first used to isolate each single root within a sector and then, using the bisection method, the formant is obtained to a prescribed level of accuracy. It should be emphasized that the ability to specify a relatively coarse formant resolution is a significant advantage, and should be contrasted to standard root-finding techniques where each complex root pair must be estimated accurately in order to carry out the deflation of the polynomial prior to the extraction of the next pair of roots.

C. Theoretical Background

The method presented in this paper is based on a number of concepts from the theory of complex variables. Recall that for any complex number z we have the polar representation $z = |z|e^{j\theta} = |z|e^{j\arg z}$. The function $\arg z$ is multivalued in z, each value differing by a multiple of 2π. An important quantity with regard to this article is the change in $\arg z$ as z moves along a curve Γ. We denote this change by $[\arg z]_\Gamma$. If a closed curve Γ circles the origin $n(\Gamma)$ times, it is easy to see that $[\arg z]_\Gamma = 2\pi n(\Gamma)$. The integer $n(\Gamma)$ is known as the winding number of the curve Γ. Using Cauchy's integral formula, $n(\Gamma)$ can be expressed as

$$n(\Gamma) = \frac{1}{2\pi i} \int_\Gamma \frac{dz}{z}.$$

An elementary result of complex analysis can be used to express the number of zeros enclosed by the closed curve Γ, counting multiplicity, of the analytic function $f(z)$ as the contour integral

$$\frac{1}{2\pi i} \int_\Gamma \frac{f'(z)}{f(z)} dz.$$

Making the transformation $w = f(z)$, it is clear that

$$\frac{1}{2\pi i} \int_\Gamma \frac{f'(z)}{f(z)} dz = \frac{1}{2\pi i} \int_{f(\Gamma)} \frac{dw}{w}.$$

Hence, the number of zeros of the function $f(z)$ enclosed by the curve Γ is the number of times that the curve $f(\Gamma)$ circles the origin, that is, $n(f(\Gamma))$ the winding number of $f(\Gamma)$. For more details, see [5].

In our case $f(z) = P(z)$, the polynomial obtained from the LPC analysis and Γ is the perimeter of a sector of the unit circle. Fig. 1 illustrates both Γ and $P(\Gamma)$ for two different sectors containing one and two roots, respectively. In Fig. 1(a), $P(\Gamma)$ circles the origin once indicating that one root of $P(z)$ lies within the corresponding sector. Fig. 1(b) illustrates the case where the sector contains two roots. Hence, the problem of determining the number of roots in a sector Γ of the unit circle has been reduced to determining the number of times that the curve $P(\Gamma)$ circles the origin. The method described in the next section was motivated by an algorithm given in [5].

D. Numerical Implementation

Given a predictor polynomial $P(z)$ and a closed curve Γ consisting of a sector of the unit circle determined by rays at angles $0 < \theta_1 < \theta_2 < \pi$, the winding number can be obtained numerically by parameterizing the curve $\Gamma(\xi) : 0 \le \xi \le 1$ and choosing a sequence of points $0 \le \xi_1 \le \xi_2 \le \cdots \le \xi_n = 1$ that are sufficiently dense that

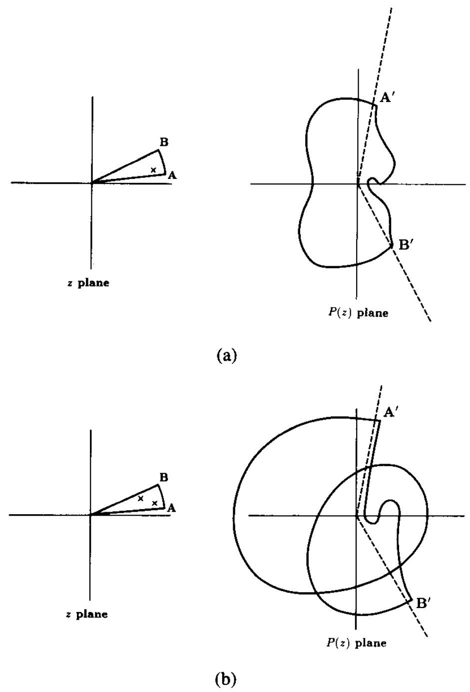


Fig. 1. Γ and $P(\Gamma)$. The two curves Γ and $P(\Gamma)$ are illustrated for two different cases. (a) one root in the sector bounded by Γ. (b) two roots in the sector bounded by Γ.

$P(\Gamma)$ can be approximated by $P(\Gamma)'$ – the curve consisting of the sequence of straight line segments joining $P(\Gamma(\xi_i))$ and $P(\Gamma(\xi_{i+1}))$. The approximation need only be close enough that the winding number $n(P(\Gamma)) = n(P(\Gamma)')$.

For the boundary of a sector bounded by the unit circle and the rays $\theta = \theta_1$ and $\theta = \theta_2$, the obvious partitions are as follows:

$$\theta = \theta_1 : z_i = a_i e^{j\theta_1}, \qquad 0 \le a_1 \le \cdots \le a_{m_1} = 1$$
$$\theta = \theta_2 : z_i = b_i e^{j\theta_2}, \qquad 0 \le b_1 \le \cdots \le b_{m_2} = 1$$
$$\text{arc} : z_i = e^{j[\theta_1 + c_i(\theta_2 - \theta_1)]}, \quad 0 \le c_1 \le \cdots \le c_{m_3} = 1.$$

In order to ensure that the sequence $\{z_i\}$ is sufficiently dense, the following criterion is used. First, the complex plane is partitioned into the octants:

$$C_k = \left\{ z | k\frac{\pi}{4} \le \arg(z) < (k+1)\frac{\pi}{4} \right\}, \quad \text{for } k = 0, 1, \cdots, 7.$$

The parameterization of the ray involves the selection of an initial partition of each [0, 1] interval and a refinement of the partition so that the values $P(z_i)$ and $P(z_{i+1})$ lie in adjacent octants. Although it is impossible to ensure that no complete circles about the origin take place undetected between z_i and z_{i+1}, it is clear that such an event can be prevented if enough points are chosen for the original partitions of the [0, 1] intervals.

Once sufficiently fine partitions of the three curves that constitute the sector boundary have been obtained, the estimated winding number of $P(\Gamma)$ (and hence the number of roots enclosed by the curve) is calculated by recording the two values:

$$N_+ = \text{the number of transitions from region } C_7 \text{ to region } C_0$$
$$N_- = \text{the number of transitions from region } C_0 \text{ to region } C_7.$$

as the curve is traversed in a counter clockwise direction. The estimate of $n(P(\Gamma))$ is the difference $n(P(\Gamma)') = N_+ - N_-$.

The efficiency of the algorithm can be improved dramatically by using an analytic estimate for the change in $\arg z$ along the arc. The modification is based on the observation that

$$P(z) = z^m + a_{m-1}z^{m-1} + \cdots + a_1 z + a_0 \approx z^m$$

when $|z|$ is large. Therefore, on an arc of radius $R \gg 1$ as arg z goes from θ_1 to θ_2, $P(z)$ goes from

$$P(Re^{j\theta_1}) = R^m e^{jm\theta_1} \text{ to } P(Re^{j\theta_2}) = R^m e^{jm\theta_2}$$

and the change in argument can be calculated directly as

$$\Delta \arg = m(\theta_2 - \theta_1). \tag{7}$$

Hence, by replacing the arc with radius $R = 1$ by one with large radius, it is possible to avoid all polynomial evaluations on the arc. It should be noted that, since all of the roots for a stable filter must lie within the unit circle, the replacement of an arc unit radius by one at larger radius does not affect the root count for the sector.

Since much of the "circling" of the origin by the curve $P(\Gamma)$ takes place on the arc, this analytic estimate given by (7) saves a large number of polynomial evaluations. The only drawback is that the use of an arc of radius $R > 1$ requires that the parameterization of each of the rays be extended out to that radius as well. In practice we have found that an arc of radius $R = 2$ serves very well. As all roots of $P(z)$ lie within the unit circle, few polynomial evaluations are needed to extend the partition from $R = 1$ to $R = 2$.

Since only the rays need be processed numerically, the algorithm can be formulated to allow the separate processing of each ray. This is done by constructing a function $N(\theta)$ with the property that the number of roots in the sector bounded by the rays θ_1 and θ_2, with $\theta_1 \le \theta_2$, is given by $N(\theta_1) - N(\theta_2)$.

To define $N(\theta)$ we assume that a ray has been chosen at angle θ and that a parameterisation for this ray is given in terms of the parameter t. For any real value t with $t \ge 0$, let $C(t)$ denote the region containing the point $P(te^{j\theta})$. If $t_i, i = 0$ to $i = M$ is a partition of the ray, we let $\rho(t_i, t_{i+1})$ denote the number of octants (modulo 8) between $C(t_i)$ and $C(t_{i+1})$. The definition of $N(\theta)$ then proceeds as follows.

1) Starting with the initial partition of the interval [0, 2] given by [0, 0.2, 0.4, 0.6, 0.8, 1.0, 2.0], refine the partition to a sequence $[t_0, t_1, \cdots, t_M]$ with the property that $\rho(t_i, t_{i+1}) \le 1$. The partition is refined by bisecting the interval on which $\rho > 1$.
2) For the sequence $\{t_i\}$, let $N_1(\theta) = N_+ - N_-$ as the ray is traversed from t_0 to t_M.

TABLE I
AN EXAMPLE OF A PARTITION OF $P(\Gamma)$. THE VALUES OF $\{t_i, C(t_i)\}$ ARE GIVEN FOR TWO RAYS CORRESPONDING TO THE SECTOR FROM 420 TO 1890 Hz. $P(z)$ IS A TENTH-ORDER LINEAR PREDICTION POLYNOMIAL DERIVED FROM SPEECH DATA

On the Ray $\theta_1 (F = 420$ Hz)

t_i	0.0	0.2	0.4	0.438	0.441	0.444	0.8	0.9	1.0	1.063	1.125	1.5	2.0
$C(T_i)$	0	7	6	7	0	1	0	7	6	5	4	3	3

On the Ray $\theta_2 (F = 1840$ Hz)

t_i	0.0	0.1	0.2	0.4	0.6	0.863	0.872	0..875	0.9	1.0	2.0
$C(T_i)$	0	7	6	5	4	3	2	1	0	0	7

3) Define $N(\theta) \equiv N_1(\theta) + [\![m\theta/2\pi]\!]$, where $[\![\ \ ]\!]$ denotes the greatest integer function.

For the sector bounded by the rays corresponding to θ_1 and $\theta_2, n(P(\Gamma)')$ is the sum of $N_+ - N_-$ on the rays and arc which form the boundary. On the ray given by $\theta_1, N_+ - N_- = N_1(\theta_1)$. Since the ray θ_2 is traversed from t_M to $t_0, N_+ - N_- = -N_1(\theta_2)$. From (7), the analytic estimate of the contribution from the arc is

$$\frac{\Delta \arg}{2\pi} = \frac{m(\theta_2 - \theta_1)}{2\pi}.$$

Since we are interested only in the number times that the origin is circled, we have

$$n(P(\Gamma)') = N_1(\theta_1) - N_1(\theta_2) + \left[\!\!\left[\frac{m(\theta_2 - \theta_1)}{2\pi}\right]\!\!\right]. \quad (8)$$

For any two real numbers $a \geq b \geq 0, [\![a - b]\!] = [\![a]\!] - [\![b]\!]$, hence we have

$$\begin{aligned} n(P(\Gamma)') &= N_1(\theta_1) - N_1(\theta_2) \\ &\quad + \left[\!\!\left[\frac{m\theta_2}{2\pi}\right]\!\!\right] - \left[\!\!\left[\frac{m\theta_1}{2\pi}\right]\!\!\right] \\ &= N(\theta_1) - N(\theta_2). \end{aligned}$$

At this point it is worth emphasizing that the contribution to $n(P(\Gamma)')$ from each of the two rays has been isolated.

As an illustration, consider the case of a sector defined by $\theta_1 = 0.26389$ and $\theta_2 = 1.18752$ (corresponding to formant frequencies 420 and 1890 Hz, respectively for data sampled at 10 kHz). The partitions generated for a particular 10th order linear prediction polynomial along the two rays are given in Table I. Fig. 2 shows $P(\Gamma)$ and the points $\Gamma(\xi_i)$.

For the ray θ_1, there are two transitions from octant 0 to octant 7 and 1 transition from octant 7 to octant 0 so $N_1(\theta_1) = 1 - 2 = -1, m\theta_1/2\pi = 0.42$ and $N(\theta_1) = -1. - [\![0.42]\!] = -1$. Similarly, for the ray θ_2, there are two transitions from octant 0 to octant 7 and no transitions from octant 7 to octant 0 so $N_1(\theta_2) = -2, m\theta_2/2\pi = 1.89$, and $N(\theta_2) = -2 - 1 = -3$. Hence, the number of roots in this sector is

$$\begin{aligned} n(P(\Gamma)) \approx n(P(\Gamma)') &= N(\theta_1) - N(\theta_2) \\ &= -1 - (-3) = 2. \end{aligned}$$

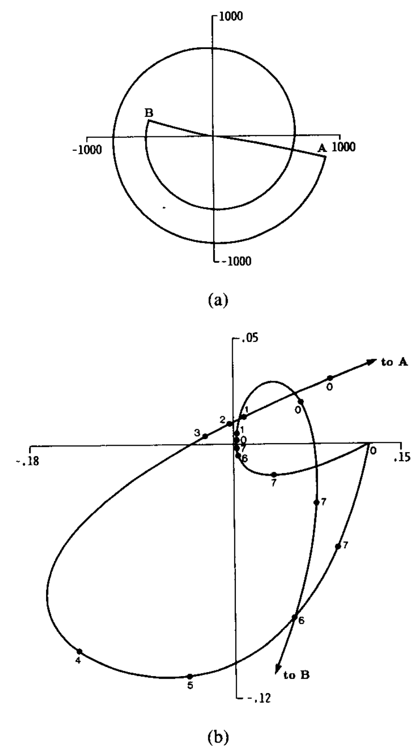


Fig. 2. $P(\Gamma)$ for a particular tenth-order polynomial. (a) The entire curve $P(\Gamma)$ curve is given with $-1000 \leq \text{Re}(P(z)) \leq 1000$ and $-1000 \leq \text{Im}(P(z)) \leq 1000$. (b) A blow-up of the curve $P(\Gamma)$ near the origin is shown. The numbers on the two curves correspond to those given in Table I.

III. TEST RESULTS

It is worthwhile at this point to examine the computational considerations involved in using the method outlined in Section II-D to determine the number of formants in a given range of frequencies. The fundamental operation is the evaluation of the polynomial $P(z)$ at a sequence of points on the sector boundary. The procedure is made particularly attractive by the fact that $P(z)$ need only be calculated accurately enough to place it in the correct octant C_k. In fact, the magnitude of $P(z)$ is of no consequence. Even an error in $P(z)$ that places it in the wrong octant will only result in an error if N_- or N_+ is affected. The test results described in Section III-A, indicate that the average number of polynomial evaluations required per sector is approximately 15 so the computational demands of the algorithm are low in operation count and each evaluation does not require a very high level of numerical precision. These computational features of the algorithm make it well suited for implementation in fixed-point arithmetic on any of a number of widely used specialized DSP chips. The results obtained from one such implementation on a Texas Instrument TMS320C25 DSP chip are given below.

It must be emphasized that the algorithm presented above is not well suited as a general method of finding roots of polynomials. It is easy to see that the number of polynomial evaluations will be large whenever multiple roots are located within a small sector. However, for polynomials obtained from LPC analysis the following sections show that the method works well.

A. Formant Estimation

The data used to test the algorithm described in Section II-D were obtained through the pitch synchronous analysis of speech data from a number of speakers. A total of 1613 frames of data were used consisting of seven utterances recorded by six different speakers. The data used were sampled at 10 kHz and stored in a 12-bit data format. For the seven utterances, three were analyzed using the covariance technique and four using autocorrelation. The data in four cases was pre-emphasized and in the other three it was not. For five of the analyses, a window was applied and in the remaining two there was no windowing. In all cases the LPC filter order was 10. The data represents a mixture of the results that would have been obtained through a typical application of LPC data analysis.

The algorithm as described in Section II-D can be used directly to determine the number of formants in any specified frequency range. In addition, by using the bisection technique, formant frequencies can be estimated to any prescribed level of accuracy. An implementation of the algorithm in Fortran using single precision arithmetic was used to test the accuracy of the method both to determine the number of formants in a given frequency range and to locate formant frequencies to a prescribed accuracy. In each case, the results were compared to the formant frequencies obtained by using a double-precision implementation of Muller's method [11] with deflation to determine roots of the LPC polynomials for each of the 1613 frames of test data.

To test the ability of the algorithm to estimate the number of formants in a specified frequency range, the test data described above was processed repeatedly using randomly generated formant ranges and the number of formants determined by the algorithm to lie in each range was compared to the number obtained by using root finding. Statistics relating to the number of polynomial evaluations required for each application of the algorithm were also gathered.

In more than 170 000 random trials, the algorithm produced only 10 errors in the estimated number of formants. The average number of polynomial evaluations per trial was 14.78 indicating a very low computational load for the algorithm. It should be noted that, of the 10 errors that occurred, the location of the formant that caused the error was within 1 Hz (0.000628 rad) of a sector boundary in six cases and within 5 Hz (0.00314 rad) in all 10 cases.

The same Fortran implementation was used in conjunction with the bisection technique to determine all formants for each data frame. Here, the first step is to determine a sector bounded by two rays θ_1 and $\theta_2 (\theta_1 < \theta_2)$ such that the sector contains exactly one root. This means that $N(\theta_1) = N(\theta_2)+1$. Hence, the sub-sector containing the root can be obtained by comparing $N((\theta_1 + \theta_2)/2)$ with $N(\theta_1)$ and $N(\theta_2)$. A significant advantage of this approach over root finding is the ability to halt the bisection at any point when the required accuracy has been achieved.

The frequency range, 150–4500 Hz, which was examined encompasses all generally prescribed formant frequency ranges for 10-kHz data (see [8], [9], and [1]). Bisection was carried out until all formants were located to within 1 Hz.

The 1613 test frames contained 6820 formants for an average of 4.23 formants per frame. Only five of the 6820 formants were not located to within the prescribed accuracy. Of these errors, three were between 5 and 10 Hz, and two were between 10 and 20 Hz. The average error was 0.553 Hz and the average number of polynomial evaluations per frame was 537.9 (126.22 evaluations per formant) indicating that the algorithm is very efficient.

The algorithm was coded in assembly language to run on a 10-MHz TMS320C25 signal processor. Again, bisection was used to locate formants within 1 Hz and the average number of evaluations per frame was 530.2 For the fixed-point implementation, the average error was 1.13 Hz with an error rate that decreases as follows: 215 formant errors (3.15%) greater than 5 Hz; 83 errors (1.22%) greater than 10 Hz; and 29 errors (0.43%) greater than 20 Hz.

The execution speed of the algorithm depends on the formant range selected and on the required resolution. Test speeds varied from 52 frames per second using a 50–4950 Hz range and 1 Hz resolution to 161 frames per second with a range of 1000–3000 Hz and 10 Hz accuracy.

Parsons [10] uses a specific eighth-order polynomial with formants and bandwidths (assuming a sampling frequency of 10 kHz) at 475 Hz (Bandwidth = 25 Hz), 1850 Hz (40 Hz), 1950 (50 Hz) and 3150 Hz (100 Hz) to examine methods for resolving closely spaced formants. For this example, the second and third formants cannot be resolved by peak picking but may be obtained using the pole-enhancement methods proposed by McCandless [8] and Kang and Coulter [6]. Using the algorithm outlined in this paper, the four formants were located with a maximum error of 0.5 Hz. A total of 548 polynomial evaluations was required.

B. Bandwidth Estimation

While many applications require only estimates of formant locations, some require, in addition, estimates of the corresponding 3-dB bandwidths. For a complex root $z = r_0 e^{j\theta_0}$ of the LPC polynomial, the bandwidth estimate provided by (2) requires the determination of the root modulus r_0. The obvious approach is to conduct a line search along the ray $\theta = \theta_0$ to determine the radius at which $f(r) = |P(re^{j\theta_0})|$ is minimized. This method was applied to the same data that was analyzed in the last section.

The bandwidth estimation algorithm was implemented in Fortran with single precision floating point arithmetic. Formant estimates were obtained to the same 1-Hz precision as were the results of the previous tests. Bandwidths were determined for all 6820 formants by using a simple line search [3].

For the data that was processed, the behaviour of the function $f(r)$ was characterized by two patterns. In most instances there was a single minimum but, for some of the data, there were two minima, one near $r = 1$ and the other closer to $r = 0$. To locate the global minima of $f(r)$ the interval [0, 1] was divided in three parts and the above line search was applied to each part.

The line search required an average of 45 function evaluations per formant to estimate the root modulus and hence the formant bandwidth. The average error was 0.95 Hz. These results indicate that a line search provides a good bandwidth estimate using an algorithm that is easily implemented in fixed point arithmetic and increases the computational load by approximately 35% following formant estimation.

For this data an average of about 175 polynomial evaluations were required to locate a formant and determine its bandwidth. The 6820 formants and bandwidths can be found using Muller's method to find all the roots of each polynomial at a cost of about 45 polynomial evaluations per formant. Although Muller's method requires about one quarter the number of polynomial evaluations, it must be implemented in relatively high precision in order to carry out the deflation of the polynomial.

IV. Conclusions

In this paper, we have discussed the estimation of formant frequencies and bandwidths using the roots of the prediction polynomial derived from linear prediction analysis of a speech waveform. The relationship between root location and formant/bandwidth estimates (3) and (4) was presented and compared with the formula most frequently cited in the paper, (1) and (2). The difference was shown to be small for roots near the unit circle. An examination of results derived from actual speech data indicate that a large percentage of the roots associated with formants are located near to the unit circle. Hence, we conclude that it is justified to use (1) and (2) in practice.

In addition, a new technique of determining formant locations was presented. The method determines the number of roots in any given sector of the unit circle by numerically estimating the winding number $n(P(\Gamma))$ where $P(z)$ is the prediction polynomial and Γ is the boundary of the sector. In contrast to most root finding methods, there is no need to determine each root accurately. The ability to specify a coarse tolerance in resolving formant locations makes the algorithm particularly well suited for real-time processing of speech data. The algorithm is computationally efficient requiring an average of approximately 15 polynomial evaluations per sector for data derived from a tenth-order LPC analysis. As well, the computations involved can be easily implemented in fixed point arithmetic with little loss in accuracy, allowing high-speed implementations of the algorithm on standard fixed-point DSP hardware.

References

[1] C. Gunnar M. Fant, "On the predictability of formant levels and spectrum envelopes from formant frequencies," in H. Lunt M. Halle and H. MacLean, Eds., *For Roman Jakobson*, 1956, The Hague, The Netherlands: Mouton, chap. 3, pp. 44–56.

[2] J. L. Flanagan, *Speech Analysis Synthesis and Perception*. New York: Springer-Verlag, 1972.

[3] G. E. Forsythe, M. A. Malcolm, and C. B. Molar, *Computer Methods for Mathematical Computations*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[4] Editor Geoff Bristow, *Electronic Speech Recognition*. New York: McGraw-Hill, 1986.

[5] P. Henrici, *Applied and Computational Complex Analysis*. Volume 1. New York: Wiley, 1974.

[6] G. S. Kang and D. C. Coulter, "A 600 bits/second voice digitizer (linear predictive formant vocoder," Tech. Rep., U.S. Naval Res. Lab., 1976.

[7] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. New York: Springer-Verlag, 1976.

[8] S. S. McCandless, "An algorithm for automatic formant extraction using linear prediction spectra," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 135–141, 1974.

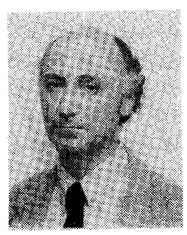
[9] R. J. Niederjohn and M. Lahat, "A zero-crossing consistency method for formant tracking of voiced speech in high noise levels," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 349–355, 1985.

[10] Thomas Parsons, *Voice and Speech Processing*. New York: McGraw-Hill, 1986.

[11] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipies*. Cambridge, UK: Cambridge Univ. Press, 1986.

[12] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.

[13] Roy C. Snell and B. C. Dickson, "Investigation of a speaker verification system using the parameters derived from CRC vocoder," Tech. Rep., Centre for Speech Technology Res., Victoria, B.C., Canada, 1983.



Fausto Milinazzo received the B.Sc. degree in mathematics and physics in 1970 and the Ph.D. degree in applied mathematics in 1974 from the University of British Columbia, Vancouver, B.C., Canada.

Currently he is professor of mathematics at Royal Roads Military College, Victoria, B.C. His interests include numerical analysis and computational fluid mechanics.

Dr. Milinazzo is a member of the Society for Industrial and Applied Mathematics.

Roy C. Snell (M'82) received the B.Sc. degree in Mathematics and Physics and the M.Sc. degree in mathematics from Queen's University, Kingston, Ont., Canada in 1965 and 1966, respectively, and the Ph.D. degree in mathematics from the University of British Columbia in 1972.

Since that time, he has been a member of the Mathematics Department at the Canadian Military College (Royal Roads) in Victoria, British Columbia where he is currently an Associate Professor. His research interests lie in the area of signal processing with a particular emphasis on speech processing and neural networks. He has worked on algorithm design in the areas of speech parameter extraction, residual excited LPC synthesis, audio pattern matching, and speaker verification. He was an associate in the Centre for Speech Technology Research at the University of Victoria from 1983 to 1990 and is currently the Vice-President of its successor organization, Speech Technology Research Ltd.