LOG AMPLITUDE MODELING OF SINUSOIDS IN VOICED SPEECH

Najam Malik and W. Harvey Holmes

School of Electrical and Telecommunications Engineering, The University of New South Wales,
Sydney 2052, Australia.

N.Malik@ee.unsw.edu.au, H.Holmes@unsw.edu.au

ABSTRACT

We present an algorithm for all-pole (envelope) modeling of the amplitudes of sinusoids present in voiced speech segments which works even when the number of sinusoids is very small, as occurs with high-pitched speakers. In contrast to previous methods, this algorithm minimizes a squared error criterion in the log amplitude domain rather than the amplitude domain, and so is better matched to the properties of the human auditory system. A weighted iterative approach is used to get near optimal solutions to this otherwise nonlinear problem. This new frequency domain log amplitude modeling (LAM) algorithm gives impressive results, especially in the case of high pitched female voices where conventional linear prediction methods are inadequate. The algorithm can easily be generalized to develop pole-zero models.

1. INTRODUCTION

Spectral amplitudes of sinusoids present in voiced speech frames display a very large dynamic range that naturally requires a logarithmic scale for adequate representation. In addition perceptual characteristics of the human auditory system seem to closely follow a logarithmic scale. Based on these considerations it is reasonable to use an error measure which is the sum of squared differences, on a logarithmic amplitude scale, between a given set of amplitudes and the corresponding magnitude response of an all pole model:

$$J = \sum_{\omega\in\Omega} \left[\log D_r(\omega) - \log \left|H\left(e^{j\omega}\right)\right|\right]^2 . \qquad (1)$$

In (1) Ω denotes a discrete set of frequencies ω where the real-valued amplitude data $D_r(\omega)$ are specified. Also, $H(e^{j\omega})$ is the frequency response of the desired all-pole model

$$H(z) = \frac{G}{A(z)} = \frac{G}{1 + a_1 z^{-1} + \cdots + a_p z^{-p}} , \qquad (2)$$

where a_i are the real-valued model parameters, p is the order of the model and G is a constant gain.

The error measure in (1) directly addresses spectral envelope errors, unlike the discrete all–pole (DAP) modeling method [2], which minimizes the discrete Itakura-Saito measure, and the minimum variance distortionless response (MVDR) method [6], which results in all-pole models that are a type of average of all lower order all-pole models.

Since it is defined in terms of logarithmic amplitudes (and can easily be expressed in terms of decibels), this error measure is also more relevant to the human auditory system than other common mean square error measures based on simple amplitudes.

The minimization of J is a nonlinear problem whose solution generally requires optimization techniques. In order to avoid the use of these techniques we consider an iterative procedure that, at the mth step, minimizes the error measure

$$J^m = \sum_{\omega\in\Omega} \left[\frac{\left[\log D_r(\omega) - \log\left|G^{m-1}/A^{m-1}\left(e^{j\omega}\right)\right|\right]^2}{\left|D(\omega) A^{m-1}\left(e^{j\omega}\right) - G^{m-1}\right|^2} \times \left|D(\omega) A^m\left(e^{j\omega}\right) - G^m\right|^2 \right] \qquad (3)$$

In equation (3) the superscript m indicates the iteration number and $D(\omega)$ is a complex valued function whose magnitude is the same as that of the real data $D_r(\omega)$ for $\omega \in \Omega$. At each step the minimization of J^m with respect to the a_i^m and G^m is linear in these unknowns. If we start with an appropriate set of values for these parameters and the sequence of values converges then it is clear that J^m approaches the original error measure J. Similar linearization techniques, without the use of the log function, have been discussed in statistical signal processing by Mullis and Roberts [5], and in control systems by Sanathan and Koerner [9]. Our approach below follows the work of Kobayashi and Imai [3].

2. DESCRIPTION OF THE ALGORITHM

The first issue concerns the specification of Ω. For reasons of computational efficiency we will take frequencies that are uniformly spaced around the unit circle: $\Omega = \{\omega_k = 2\pi k/N\text{: } k = 0, 1, \dots, N-1\}$, with N even. This choice of Ω will enable us to use the DFT in the calculations that follow.

In harmonic coders the real-valued amplitudes $D_r(\omega)$ are obtained by examining the peaks in the short time Fourier transform of a frame of speech. For high-pitched speech segments, especially from female speakers, this set provides a very sparse sampling of the envelope of the spectral amplitudes. With the specification of Ω above we need to interpolate between these spectral envelope samples. A natural and computationally simple first step is to use linear interpolation. More numerically intensive interpolation methods based on cubic splines can also be used [1].

The phase of the function $D(\omega)$ in (3) can be specified without any constraint. However, since linear prediction methods often give rise to all-pole models with the minimum phase property, we similarly require that $D(\omega)$ be minimum phase. This minimum phase sequence can be uniquely obtained [7] from the interpolated magnitude data using cepstral coefficients,

$$d_n = \frac{1}{N}\sum_{k=0}^{N-1} \log D_r(2\pi k/N) e^{j2\pi kn/N}, \qquad (4)$$

$n = 0, \dots, N-1$, and

$$\tilde{d}_n = \begin{cases} 2d_n, & n = 1, \dots, N/2-1 \\ d_n, & n = 0, N/2 \\ 0, & n = N/2+1, \dots, N-1\,. \end{cases} \qquad (5)$$

With the definitions in (4) and (5) the minimum phase sequence $D(\omega_k)$ is given by

$$\log D(2\pi k/N) = \sum_{n=1}^{N-1} \tilde{d}_n e^{-j2\pi kn/N}\,. \qquad (6)$$

Next we write the error measure in (3) in matrix-vector notation. For this purpose we define the following weight function for the mth iteration

$$W^m(\omega_k) = \frac{\left[\log D_r(\omega_k) - \log\left|G^{m-1}/A^{m-1}\left(e^{j\omega_k}\right)\right|\right]^2}{\left|D(\omega_k)A^{m-1}\left(e^{j\omega_k}\right) - G^{m-1}\right|^2}, \qquad (7)$$

where $\omega_k = 2\pi k/N$, $k = 0, 1, \dots, N-1$. We also define the vector $\mathbf{a}^m = \begin{bmatrix} 1 & a_1^m & a_2^m & \dots & a_p^m \end{bmatrix}^{\mathrm{T}}$, where T stands for matrix transpose. With this notation the error measure for the mth iteration in (3) becomes

$$J^m = \mathbf{a}^{m^{\mathrm{T}}}\mathbf{A}\mathbf{a}^m - G^m\mathbf{c}^{\mathrm{T}}\mathbf{a}^m - G^m\mathbf{a}^{m^{\mathrm{T}}}\mathbf{c} + \left(G^m\right)^2 g\,. \qquad (8)$$

In equation (8) the $(p+1)\times(p+1)$ matrix $\mathbf{A}$ is symmetric and Toeplitz with the uvth element given by

$$\alpha_{uv} = \sum_{k=0}^{N-1} W^m(2\pi k/N)\left|D(2\pi k/N)\right|^2 e^{j2\pi|u-v|k/N}, \qquad (9)$$

where $u, v = 1, 2, \dots, p+1$. The (column) vector $\mathbf{c}$ has uth component

$$c_u = \sum_{k=0}^{N-1} W^m(2\pi k/N) D(2\pi k/N) e^{j2\pi uk/N}, \qquad (10)$$

with $u = 1, 2, \dots, p+1$. Note that $\mathbf{A}$ and $\mathbf{c}$ can each be computed by a single application of the DFT. Finally the scalar g is given by

$$g = \sum_{k=0}^{N-1} W^m(2\pi k/N)\,. \qquad (11)$$

The expression for J^m in (8) can be further simplified to the following quadratic form

$$J^m = \begin{bmatrix} \mathbf{a}^{m^{\mathrm{T}}} & G^m \end{bmatrix} \begin{bmatrix} \mathbf{A} & -\mathbf{c} \\ -\mathbf{c}^{\mathrm{T}} & g \end{bmatrix} \begin{bmatrix} \mathbf{a}^m \\ G^m \end{bmatrix}. \qquad (12)$$

Minimization of J^m with respect to the vector of parameters $\mathbf{a}^m$ and the gain G^m gives rise to a system of linear equations in these unknowns and the minimum value of J^m:

$$\begin{bmatrix} \mathbf{A} & -\mathbf{c} \\ -\mathbf{c}^{\mathrm{T}} & g \end{bmatrix} \begin{bmatrix} \mathbf{a}^m \\ G^m \end{bmatrix} = \begin{bmatrix} J_{\min}^m \\ \mathbf{0} \end{bmatrix}. \qquad (13)$$

Since $\mathbf{A}$ is symmetric and Toeplitz, the coefficient matrix of this system of equations has a special structure which leads to a fast recursive algorithm for the solution [4]. This solution is used to update the weight function in (7) and the process is repeated until improvements in the error measure are not significant. At the first iteration of the algorithm the value of the weight function can simply be set to 1. That is, take $W^1(\omega_k) = 1$, with $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N{-}1$.

The complete algorithm can be summarized with the steps below:

1. Obtain a set of peaks from a windowed DFT of a frame of voiced speech.
2. Interpolate between the peaks to obtain values of the spectral envelope that are uniformly spaced around the unit circle.
3. Use (4), (5) and (6) above to obtain the minimum phase sequence corresponding to the interpolated spectral magnitude values of step 2.
4. Set the weight function to the initial value 1.
5. Compute the quantities **A**, **c**, and g using equations (9), (10) and (11).
6. Solve (13) for the parameter vector $\mathbf{a}^m$, the gain G^m, and $J^m_{\min}$.
7. If the change in $J^m_{\min}$ from $J^{m-1}_{\min}$ is below a desired threshold, then stop. Otherwise continue with the next step.
8. Update the weight function using (7).
9. Repeat starting at step 5.

The method described above can be extended to yield pole-zero models, using the same error criterion, by replacing the constant gain G in (2) with a numerator polynomial $B(z)$.

3. EXPERIMENTAL RESULTS

Convergence of the algorithm is difficult to prove analytically. However, since each of the sequence of linear problems in (13) has a unique solution, we do expect the resulting sequence of parameters to have a convergent behaviour. This is borne out by our experimental results, which show that the algorithm converges very rapidly. In fact there is no significant change in the error measure after only two iterations. Similarly there are no analytical reasons that would guarantee a stable all-pole filter. The explicit use of the minimum phase condition in step 3 above did, however, lead to stable all-pole models in all the cases that we have tested.

In figure 1 we have a typical test case of a frame of high-pitched voiced speech from a female speaker. The segment has an approximate fundamental frequency of 390 Hz, which is near the high end of the observed range of fundamental frequencies.

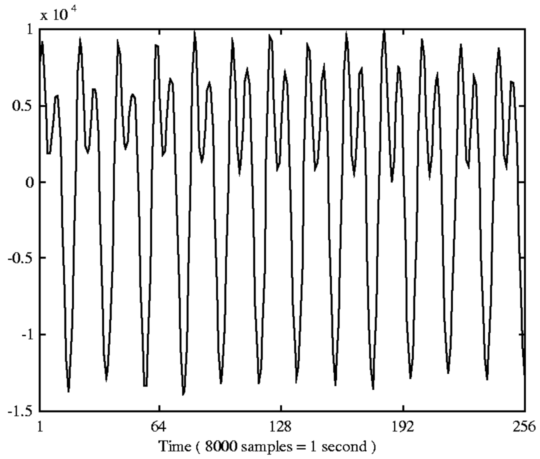

Figure 1: Voiced speech frame from a female speaker.

Figure 2 shows the magnitude of the widowed DFT, using a Hamming window, of the frame of figure 1. The peaks in figure 2 that are indicated with crosses were selected by using a SEEVOC type peak-picking algorithm [8]. This set of peaks provides only ten samples of the spectral envelope in the speech bandwidth because of the high fundamental frequency.

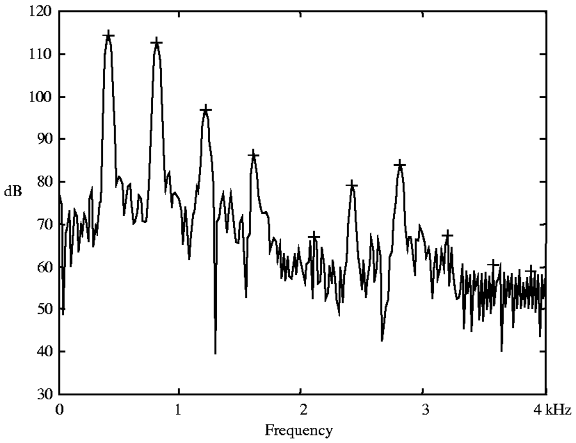

Figure 2: Windowed DFT of frame of figure 1 showing the selected peaks.

Figure 3 shows the spectrum of figure 2 together with the magnitude response of the 14th order all-pole filter obtained after two iterations with the algorithm described above. Linear interpolation between the peaks was used in step 2 for this example. For comparison, figure 4 shows the magnitude response of a 14th order all-pole filter obtained using linear prediction (auto-correlation method), superimposed on the spectrum of figure 2. The gain of this filter was adjusted to minimize the spectral distortion in dB.

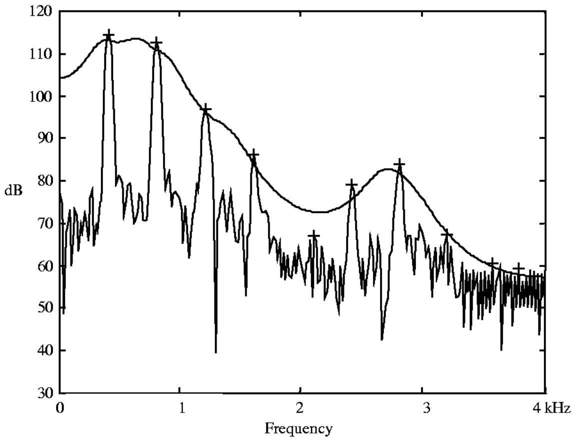

Figure 3: Magnitude response of 14th order all-pole model obtained from the algorithm of section 2 after two iterations.

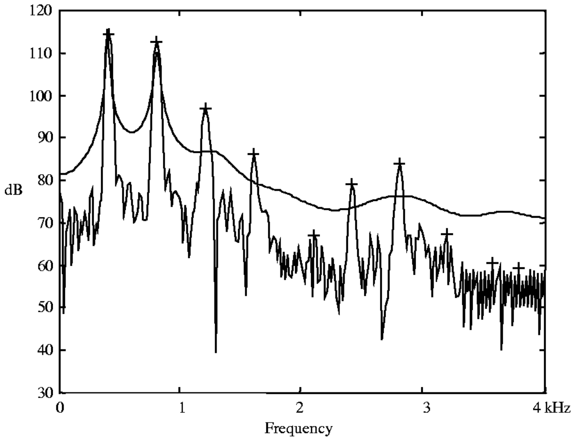

Figure 4: Magnitude response of 14th order all-pole filter obtained by using linear prediction.

The difference between the two envelopes is quite remarkable. The linear prediction envelope of figure 4 is dominated by the appearance of the two peaks corresponding to the first two harmonics. It also completely misses the low amplitude peaks at mid to high frequencies. This behaviour is quite typical and results from the sparse sampling of the spectral envelope by the high pitched periodic excitation signal, leading to modeling dominated by the major harmonics. The spectral envelope of figure 3, on the other hand, follows the trend of the spectral peaks very well across the entire speech bandwidth. This is especially true in the vicinity of the perceptually important second formant and at high frequencies, where the amplitudes are low.

4. CONCLUSIONS

We have described an algorithm for frequency domain all-pole modeling of the spectral amplitudes of sinusoids present in voiced speech frames. The algorithm works well even for cases where the fundamental frequency (pitch) is very high, which leads to the failure of previous modeling methods.

In the new algorithm the filter parameters and the gain are jointly computed to iteratively minimize the squared error between the log spectral magnitudes. Experimental results show that, after just two iterations, the algorithm gives minimum phase models that tend to closely follow the spectral peaks, even when these peaks are sparse. Finally, the algorithm also has a simple extension to pole-zero modeling

REFERENCES

[1] C. D. Covington, "Cubic spline modeling of speech spectra," *Proc. ICASSP*, vol. 3, pp. 1125-1128, 1985.

[2] A. El-Jaroudi and J. Makhoul, "Discrete all-pole modeling for voiced speech," *Proc. ICASSP*, vol. 1, pp. 320-323, 1987.

[3] T. Kobayashi and S. Imai, "Design of IIR digital filter with arbitrary log magnitude function by WLS techniques," *IEEE Trans. ASSP*, vol. ASSP-38, pp. 247-252, 1990.

[4] Y. Monden, M. Yamada, and S. Arimoto, "Fast algorithm for identification of an ARX model and its order determination," *IEEE Trans. ASSP*, vol. ASSP-30, pp. 390-399, 1982.

[5] C. T. Mullis and R. A. Roberts, "The use of second-order information in the approximation of discrete-time linear systems," *IEEE Trans. ASSP*, vol. ASSP-24, pp. 226-238, 1976.

[6] M. N. Murthi and B. D. Rao, "Minimum variance distortionless response (MVDR) modeling of voiced speech," *Proc. ICASSP*, vol. 3, pp. 1687-1690, 1997.

[7] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[8] D. B. Paul, "The spectral envelope estimation vocoder," *IEEE Trans. ASSP*, vol. ASSP-29, pp. 786-794, 1981.

[9] C. K. Sanathan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Trans. AC*, vol. AC-8, pp. 56-58, 1963.