IMPROVING FORMANT BANDWIDTH ESTIMATION BY SELECTIVE LAG WINDOWING

M. Pikri, M. F. Aou-El-Yazid & M. R. El-Ghonemy

Electronics & Communications Dept. Faculty of Engineering, Cairo University, Giza, Egypt.

ABSTRACT

Formant bandwidths are severely underestimated in LPC analysis of voiced segments with low F1/F0 ratio. Parameter quantization errors aggravate the situation and cause steady nasal sounds and high vowels to have amplitude booms. Spectral smoothing using a lag window was used as a possible remedy. It is found, however, that indiscriminate smoothing reduces the crispness of the synthesized speech. Lag windowing is introduced only for frames with low F1/F0. Detection of such frames is made using a pattern classification approach after a preliminary reduced order LPC analysis stage. Estimates of F1 and F2 are found by a closed form solution of the 4th order inverse filter polynomial. A Fisher classifier is used to detect the condition of low F1/F0. If a frame is classified as having low F1/F0 the autocorrelation vector is multiplied by a binomial window. Results show improved quality over both indiscriminate spectral smoothing and the standard autocorrelation methods.

1. INTRODUCTION

Estimation of speech parameters in LPC analysis/synthesis systems is known to suffer from large errors when voiced segments with high pitch frequencies are encountered [1]. This may result in perceptually significant distortion in the synthesized speech, usually manifested as amplitude booms or as a change in voice timber. Bandwidth underestimation of the first formant is thought to be the reason of such distortion [2]. For vocoding applications, the problem is further aggravated by quantizing the spectral parameters. Very narrow bandwidths cause extremely high sensitivities to parameter quantization [3].

Two approaches were followed to alleviate the problems arising from bandwidth underestimation. In the first approach amplitude booms are avoided by adjusting the excitation level at the synthesizer [4-5]. This, however, does not solve the problem of spectral distortion. The energy is still concentrated in the narrow-band formant, and the quantization error is not reduced as it is generated during the encoding. In the second approach the sharp resonance is flattened through spectral smoothing either by bandwidth expansion [3] or by lag windowing [2]. It is found, however, that indiscriminate spectral smoothing causes low-pitch to lose its crispness. In this paper, spectral smoothing is introduced selectively: only for frames that are susceptible to bandwidth underestimation. The selection criterion is based on formant and bandwidth estimates, which are obtained as a byproduct from the SIFT pitch algorithm at low extra computational cost.
2. EFFECT OF LAG WINDOWING

Spectral smoothing by lag windowing was introduced by Itakura [2] in the linear prediction analysis using the autocorrelation method. A lag window of finite length equal to the prediction order \( P \) was multiplied by the autocorrelation \( R_{xx}^{(n)} \). The resulting all pole spectrum has been strongly affected by the lag window properties. Itakura showed that the binomial window was more effective to eliminate the extremely sharp first formant peak without smearing the higher ones.

In the present paper, the binomial window is multiplied by the covariance coefficients to test its effect on the covariance method of analysis. Two alternatives have been tested. The first alternative is to multiply the covariance coefficient \( C_{ij} \) by \( w(i-j) \) where \( w(i) = \binom{2n}{n}^{-\frac{1}{2}} w(-i) \).

and the second alternative is to multiply \( C_{ij} \) by \( w(i) \times w(j) \), i.e. the covariance matrix is multiplied by a two dimensional window.

Synthetic speech with completely known characteristics is necessary in this test. Synthetic vowels, composed of three fixed formant frequencies and bandwidths, are synthesized at various pitch frequencies. Formant frequencies were 320, 2250, and 3000 Hz, and bandwidth were 50, 80, and 120 Hz, respectively. These synthetic vowels are chosen similar to those analyzed by Itakura [2] for the sake of comparison.

Plotting the first formant bandwidth against the pitch period, as shown in Fig. 1, gave the same curves as those obtained by Itakura. The use of the two dimensional window in the covariance analysis method results in an improvement similar to that obtained due to the addition of the lag window in the autocorrelation method. This similarity is expected due to the equality in the number of windows used in each method. This is because there is no time windowing of the signal in the covariance method. However, one multiplication for each covariance coefficient can be done if the two dimensional matrix is stored before processing.

As shown in Fig. 2, using the same equivalent bandwidth for both one-dimensional and two-dimensional windows applied in the covariance method resulted in an additional increase in the bandwidth in the case of the one-dimensional window.

These results showed that the same advantages of using the lag window in the autocorrelation method could be obtained in the covariance method. The covariance method is more suitable in pitch synchronous analysis due to the absence of time window limits. The stability of the autocorrelation method has been compensated in the lattice method by continuously checking that the absolute values of the reflection coefficients did not exceed unity. The processing time of the lattice method is also equal to that of the autocorrelation method [7].

3. SELECTIVE LAG WINDOWING

It is clear from Fig. 2, that, if lag windowing is not applied, bandwidth underestimation is centered about values of \( F_0 \) that are submultiples of \( F_1 \). Other pitch frequencies result in bandwidth overestimation. It is also clear that lag windowing biases the average bandwidth estimation error to the overestimation side. It is thus plausible to apply lag windowing only when \( F_1 \) is a multiple of \( F_0 \). Estimates of \( F_1 \) and \( F_0 \) can be obtained through the preliminary analysis carried out during the SIFT pitch estimation algorithm [7]. In this algorithm the signal is lowpass filtered to 900 Hz, leaving no more than the first two formants.

Fourth order LPC analysis is then carried out efficiently in terms of the first four autocorrelation coefficients of the filtered signal. The roots of fourth order prediction polynomial can then be

![Fig. 2 Comparison between the effects of one and two dimensional lag windowing.](image-url)
found analytically, with few extra computational steps over the standard algorithm. \( F_1 \) can then be found from the roots.

Unfortunately, although the SIFT algorithm results in a good estimate of \( F_0 \), the estimate of \( F_1 \) suffers from analysis errors at high \( F_0 \), as discussed in Section 2. The value of the measured \( F_1/F_0 \) cannot thus be used directly to decide on the application of lag windowing. To overcome this difficulty, we propose to use a pattern classification algorithm that takes the outputs of the preliminary SIFT analysis as features, and decide if lag windowing is to be employed in the complete analysis. A simple Fisher linear discriminant is proposed as the pattern classifier.

To obtain the linear discriminant coefficients the following procedure was followed. Four sets of synthetic vowels were generated with different pitch frequencies, as described in Section 2. Each set has fixed formant frequencies and bandwidths, with \( F_1 \) values between 320 Hz and 820 Hz, and \( F_2 \) values between 750 Hz and 2250 Hz. The resulting data were analyzed by the SIFT algorithm and the analysis results (\( F_0 \), \( F_1 \), \( F_2 \), \( B_1 \) and the residual error) were stored as feature records. The same speech data were then analyzed using the covariance lattice harmonics method with tenth order, and the formant frequencies and bandwidths were obtained by numerically finding the roots of the prediction polynomial. The same procedure was repeated using one-dimensional lag window with \( B_{in} = 75 \) Hz.

4. Results and Conclusions:

The proposed algorithm has been tested using two analysis/synthesis systems. The first system used the Texas Synthesizer chip 5220, while the second system used an 8-bit D/A converter together with a synthesis software. A female utterance "a satellite means a man mode moon" has been used for testing both systems, since it contains some segments with high pitch frequency and low first formant frequency. Synthesizing this utterance using the 5220 chip resulted in obvious clicks. Adding the selective lag window algorithm to the covariance analysis method, removed these clicks. The D/A-based synthesis system produced less obvious clicks when the reflection coefficients were coded in 6 bits. These clicks become hard to detect as the number of coding bits is increased to 8-bits. Again, the selective lag window algorithm removed these clicks. Thus to maintain the quality of synthesized speech while reducing the number of coding bits to 6 bits, the proposed selective algorithm is recommended.

As shown in Fig. 3, the time samples of the 6-bit coding with selective lag windowing (Fig 3.d) look alike the original samples (Fig 3.a). It can be noticed also that the extra peak appearing in the time samples of Fig 3.c has disappeared in Fig 3.d. It has been noticed also that adding the lag window to the covariance analysis method indiscriminately to all frames reduces the quality of the synthesized speech due to smearing of the formant bandwidths.

---

**Fig. 3 Effect of lag windowing and coding**

- a) Original time samples.
- b) Synthesized time samples using 16 bits coding.
- c) Synthesized time samples using 6 bits coding.
- d) Synthesized time samples using 6 bits coding with lag windowing.
A feature selection algorithm could be applied to select optimum number of features to decide which frames need the application of lag windowing. This may reduce the processing time of the selective analysis algorithm.

References


