The Use of Linear Prediction of Speech in Computer Music Applications*

JAMES A. MOORER

IRCAM, Paris F75004, France

One of the results of the science of estimation theory has been the development of the linear prediction algorithms. This allows us to compute the coefficients of a time-varying filter which simulates the spectrum of a given sound at each point in time. This filter has found uses in many fields, not the least of which is speech analysis and synthesis as well as computer music. The use of the linear predictor in musical applications allows us to modify speech sounds in many ways, such as changing the pitch without altering the timing, changing timing without changing pitch, or blending the sounds of musical instruments and voices. This paper is concerned with the fine details of the many choices one must make in the implementation of a linear prediction system and how to make the sound as clean and crisp as possible.

0. INTRODUCTION

Linear prediction is a method of designing a filter to best approximate, in a mean-squared-error sense, the spectrum of a given signal. Although the approximation gives as a result a filter valid over a limited time, it is often used to approximate time-variant waveforms by computing a filter at certain intervals in time. This gives a series of filters, each one of which best approximates the signal in its neighborhood. The uses for such a filter are manyfold, ranging from geological and seismological applications [1] to radar and sonar [2], to speech analysis and synthesis [3]–[7], and to computer music [8]–[12]. We shall concentrate here on the usage of linear prediction as a method of capturing, simulating, and applying the sounds of the human voice in high-fidelity musical contexts. Even more specifically, we will concentrate on applications using only the digital computer as the medium.

This paper reports the results of work done over the last two years in searching for ways to improve the quality of speech synthesis. The findings were determined by largely informal listening tests with trained musicians.

1. MODELING OF SPEECH

This description is taken largely from Makhoul [13]. We start by modeling the sound of the human voice as an all-pole spectrum with a transfer function given by

$$H(z) = G/A(z) \tag{1}$$

where

$$A(z) = \sum_{k=0}^{p} a_k Z^{-k}, \qquad a_0 = 1 \tag{2}$$

is known as the inverse filter, G is a gain factor, a_k are the filter coefficients, and p is the number of poles or predictor coefficients in the model. If $H(z)$ is stable (minimum phase), $A(z)$ can be implemented as a lattice filter [5] as shown in Fig. 1. The reflection (or partial correlation) coefficients K_m in the lattice are uniquely related to the predictor coefficients. For a stable $H(z)$, we must have

$$|K_m| < 1, \qquad 1 \leq m \leq p. \tag{3}$$

$H(z)$ can also be implemented as a lattice form as shown in Fig. 2, as well as a product of first- and second-order sections by factoring $A(z)$ and combining complex conjugate roots to form second-order sections with all real coefficients, as shown in Fig. 3. Finally, the filter can be implemented in direct form as shown in Fig. 4.

We are excluding for the time being models that include both poles and zeros since we have not as yet investigated a satisfactory method to compute both poles and zeros reliably.

To actually synthesize a speech sound, one must drive this filter with something. This is called the excitation, and it too must be modeled to provide a reasonable representa-

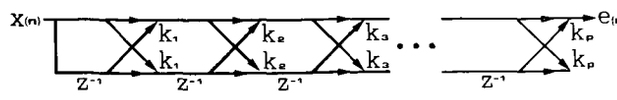

Fig. 1. Lattice form of inverse filter.

* Presented at the 59th Convention of the Audio Engineering Society, Hamburg, 1978 February 28–March 3.

tion of the speech excitation. We usually choose the excitation to be either white noise for unvoiced sounds, a wide-band pulse train for voiced sounds, or silence (for silence), as shown in Fig. 5, although this simplification should be discussed further.

Thus to summarize, if we wish to synthesize speech, the process from analysis to synthesis might be the following:

1) Extract the pitch of the original sound.

2) Compute the linear prediction coefficients for the original sound at selected points in time.

3) Decide at each point in time whether the signal is voiced, unvoiced, or silence.

4) Compute the gain factor for the original sound.

5) Create an excitation from the pitch and the voiced/unvoiced/silence decision.

6) Scale it with the computed gain factor.

7) Filter it with the computed predictor coefficients.

This is roughly the outline of the process from start to finish, but the order is not necessarily rigid. For example, we may decide not to compute a gain factor, but merely to scale the energy of the synthesized signal to correspond to the original energy in the signal.

Readers wishing to know more about the subject of linear prediction of speech should refer to the literature [6], [7].

There are numerous other decisions to be made, such as choosing a method for doing each of these things, choosing the order of the filter, deciding what form the filter should be in, how to interpolate the parameters between. We will attempt to comment on each of these.

2. ON MUSICAL APPLICATIONS

The most usual application of this technique is with respect to speech communication. The idea there is to reduce the data involved in the transmission of speech. Indeed, using linear prediction, one can quantize the various parameters and obtain striking reductions in the amount of data involved [7]. For musical purposes, however, we cannot generally afford the loss of quality implied by this quantization. Although even in speech communication the quality is important, it is not as critical as in the case of music production. At each point, we must ask ourselves, "Would I pay $5.95 for a record of this voice?"

In general, there is no point in directly resynthesizing a piece of speech or singing. One could just use directly the original segment. The only point is to be able to modify the speech in ways that would be difficult or impossible for the speaker to do. These include modifications of the driving function, such as changing the pitch or using more complex signals, changing the timing of the speech, or actually altering the spectral composition. Thus we will concentrate here not only on methods that preserve the speech quality in an unmodified reconstruction, but also that are less sensitive to modification, that can preserve the quality over a wide range of modifications.

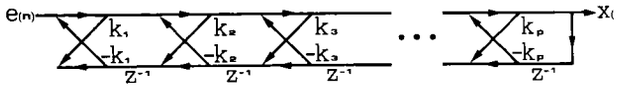

Fig. 2. Lattice form of all-pole filter. The filter is unconditionally stable if all the coefficients are of magnitude less than 1.

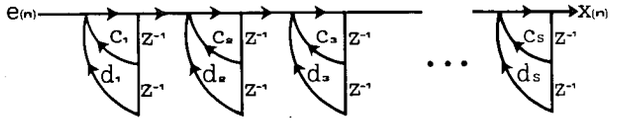

Fig. 3. Factorization of all-pole filter into second-order sections.

3. BASIC DECISIONS

The first step in the process is to detect the pitch. This is not a simple problem, but has been well studied [14]–[17]. In the musical case, we have more information beforehand than one would in the speech communication case, in that we can allow the program some amount of information about the speaker. In specific, if the range of frequencies can be bounded or identified beforehand, this eliminates immediately most of the gross errors that pitch detectors usually commit. What few gross errors remain can be corrected automatically by heuristic means. We have found that if the pitch at any given time can be limited to a range of only one octave, most of the pitch detectors reported in the literature seem to work adequately. The only question is how often should the pitch be determined. We are currently using pitch determination every 5 milliseconds, and this seems to give fine enough resolution for most purposes.

Next is the voiced–unvoiced–silence decision. This seems to be the most difficult part to automate. So difficult, in fact, that we have taken to using a graphics program to allow the composer to go through and mark the segments himself. We use a decision theoretic procedure for the initial labeling [18], [19]. We then synthesize an unmodified trial replica of the original sound. Using graphics, 150-millisecond windows of both the original and the replica are presented. The voiced–unvoiced–silence decision as determined by the computer is listed below the images. When the differences between the

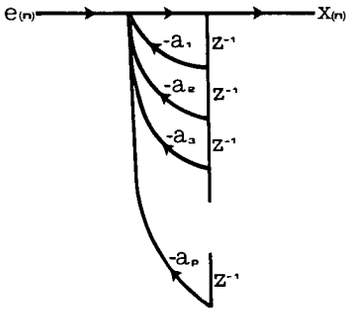

Fig. 4. Direct form for the realization of an all-pole filter.

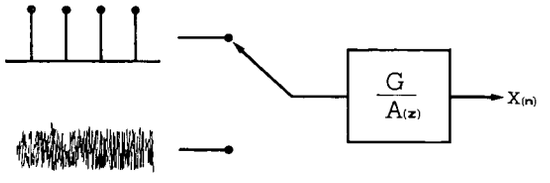

Fig. 5. Schema of the synthesis of speech using as excitation either a pulse train or white noise.

original and the replica seem to indicate an error in the decision, it is easily corrected by hand. It takes about 15 minutes to go through a 12-second segment of speech this way, which represents, for instance, about one stanza of a poem (between 35 and 40 words).

4. WHAT KIND OF PREDICTION

Usually in speech analysis, the analysis window is stepped by a fixed time, such as 10 or 15 milliseconds, and takes a fixed number of samples at each step, such as 25 milliseconds worth. This has the problem of inconsistency. A 25-millisecond window for a male voice will sometimes capture two speech pulses and sometimes three, depending on the pitch and phasing of the speech. This gives a large frame-to-frame variability in the spectral estimate. The result is a "roughness" that depends on the relation between the instantaneous pitch and the frame width.

One can decrease the effect of this phenomenon in several ways. First, by use of an all-pass filter, one may distort the phase of the speech to largely eliminate the prominance of the glottal pulse [20]. One can also use a larger analysis window so that more main pulses are incorporated and that the omission or inclusion on one pulse does not perturb the filter so strongly. Both of these remedies have the effect of blurring what are often quite sharp boundaries between voiced and unvoiced sounds. The problem is that if the analysis window overlaps significantly an unvoiced region, the extreme bandwidth of the unvoiced signals contributes to a filter that passes a great deal of high frequencies. If this filter is then used to synthesize a voiced sound, a strong buzzy quality is heard. The overall effect was that just around fricatives, the voice before and after had a strongly buzzy quality.

Another problem with using analysis windows larger than a single period is that the filter begins to pick up the fine structure of the spectrum. The fine structure is composed of those features that contribute to the excitation, notably the pitch of the sound. At the high order required for high-quality sound on wide-bandwidth original signals (we are using 55th-order filters for a deep male voice with a sampling rate of 25 600 Hz), the filter seems to capture some of the pitch of the original signal from overlapping several periods at once. The result is that even though reasonable unmodified synthesis can be obtained, the sound deteriorates greatly when the pitch is changed. This, then, is a case where the unique musical application of modification implies a more substantial change from speech communication techniques.

The solution that we adopted was the use of pitch synchronous analysis, where the analysis window is set to encompass exactly one period, and it is stepped in time by exactly one period. This prevents any fine structure representing the pitch from being incorporated into the filter itself. It also provides that in the case of the borders between voiced and unvoiced regions, no more than one period will overlap the border itself. The step size and window width is not so critical in the unvoiced portions, so we simply invent a fictitious pitch by interpolating between the known frequencies nearest the unvoiced region. There does remain a slow variation in the filters, presumably caused by inaccuracies in the pitch detection process. This can be somewhat lessened by the all-pass filter approach [20], but it does not seem to be terribly annoying in musical contexts.

Note that the adoption of pitch-synchronous analysis has implications for the type of prediction used. The most popular method is the autocorrelation method, but its necessary windowing is not appropriate for pitch-synchronous analysis. Some kind of covariance or latice method is then required. What we have chosen is Burg's method [1] because it does correspond to the minimization of an error criterion, the filter is unconditionally stable, and there exists a relatively efficient computational technique [13]. We have tried straight covariance methods with the result that the instabilities of the filters are inherent at high orders in certain circumstances and somewhat difficult to cure. One can always factor the polynomial and replace the ailing root by its inverse, then reassemble the filter, but besides being expensive, there is another reason to be discussed subsequently that is even more compelling.

There are also a number of recursive estimation techniques [21]–[23] which allow one to compute the coefficients from the previous coefficients and the new signal points. This has the advantage that no division of the signal into discrete windows is necessary. In fact, no division is possible. The problem is, again, that if the "memory" of the recursive calculation is short enough to track the rapid changes, such as from an unvoiced region to a voiced region, then it also tracks the variation of spectrum throughout a single period of the speech sound. The short-term spectrum changes greatly as the glottis opens and closes. If the memory of the calculation is long enough to smooth out the intraperiod variations, then it also tends to mix the spectra of the adjacent regions.

5. THE EXCITATION FUNCTION

To resynthesize the signal, either at the original pitch or at an altered pitch, one must synthesize an excitation function to drive the computed filters that embodies both the pitch and the voiced–unvoiced–silence decision. The most common method is to use a single impulse for each period in the voiced case and uniform noise of some sort in the unvoiced case. In the case of silence, the transient response of the filters is allowed to unwind naturally.

The problem with the single pulse is that it is not a band-limited signal. In places where the pitch is changing rapidly, this produces a roughness in the sound that is quite annoying. For this reason it is generally preferable to use a band-limited pulse of some sort [24]. One can further improve the sound by scrambling somewhat the phases of the components of the band-limited pulse to prevent the highly "peaky" appearance, but this is frosting on the cake that is not clearly perceived by most listeners. It is audible, but it is not the dramatic transformation from harsh to melifluous that one might hope. We synthesize the pulse by an inverse fast Fourier transform. This allows us to set the phases of each component independently. We

found that a slight deviation from zero phase was desirable and easily accomplished by adding a random number into the phase corresponding to −0.5 to +0.5 (radians). This range seemed sufficient to "round off" the peak. Since using the FFT is a somewhat expensive way to compute the excitation, we computed it only when the frequency changed enough that a harmonic had to be omitted or added. The synthesized driving signal was kept in a table and sampled at the appropriate rate to generate the actual excitation. This provided another benefit that we will discuss presently. Also, since recomputing the driving function using semirandom phases can give discontinuities when changing from one function to a new one, we used a raised cosine to round the ends of the driving function to zero. If a dc term is present, this is known to leave the spectrum unchanged, except for the highest harmonics, so we are assured that the driving spectrum is exactly flat up to near the maximum harmonic. The raised cosine was applied just at the beginning 10% and the ending 10% of the function.

The production of the noise for the unvoiced regions does not seem to be highly critical. We are using Gaussian noise [25].

One might ask why we attempt to synthesize the driving function. Why not use the residual of the original signal directly? This indeed has the advantage that there is no pitch detection involved and no voiced–unvoiced–silence decision at all. The problem is that then, for musical purposes, one must be able to modify the residual itself. There exist methods for doing this using the phase vocoder as a modification tool [26], [27]. There is even a recent study about making the phase vocoder more resistant to degradation from modification [28].

The problem is that to produce the residual, it is often necessary to amplify certain parts of the spectrum that might have been very weak in the original. The very definition of "whitening" the signal is to bring all parts of the spectrum up to a uniform level. There are inevitable weak parts of the spectrum—nasal zeros or some such. If there is precious little energy at a certain band of frequencies, then the whitening process will simply amplify whatever noise was present in the recording process. If this resulting noise then falls under a strong resonance when the prediction filter is applied, this filter then just amplifies that noise. The perceptual effect is that the signal looses its crispness and becomes "fuzzy," and sometimes even downright noisy.

6. CROSS SYNTHESIS

In discussing excitation functions, one must mention a particular application that has been found quite useful for the production of new musical timbres, and that is the operation of cross synthesis [10]–[12]. Here we use another musical sound as excitation, rather than attempting to model the speech excitation. What results is a bizarre but often interesting combination of the source sound and the speech sound. In this manner we can realize the sounds of "talking violin" or "talking trumpet," in a manner of speaking. In fact, if one uses the musical signal directly as excitation, quite often the result is not highly intelligible. This is because most musical signals are not spectrally flat wideband signals, but instead have complicated spectra. For this reason it is usually good to whiten the source sound. One can do this also with a low-order linear predictor as shown in Fig. 6. The error signal of a fourth- to sixth-order linear prediction process is usually sufficiently whitened to improve the intelligibility greatly, but at the expense of the clarity of the original musical source. In fact, one can choose from a continuum of sounds between the original musical source and the speech sound. Depending on the compositional goals, one might choose something more instrument-like and scarcely intelligible, progressing through stages of increasing intelligibility, or whatever.

One can use any sound as excitation with varying results. For instance, if the source sound has very band-limited spectral characteristics, such as an instrument with a very small number of harmonics like a flute, the whitening process will just amplify whatever noise happens to be present in the recording process, producing an effect somewhat like a "whispering" instrument, where the pitch and articulation of the instrument are clearly audible, but the speech sounds distinctly whispered.

One can also deliberately defeat the pitch synchronicity of the analysis to capture the fine structure of the spectrum. If one then filters a wideband sound, such as the sound of ocean waves, one can as the order increases impose the complete sound of the voice on the source. We can in this manner realize something like the sounds of the sirens on the waves, or the "singing ocean." In general this sort of effect takes a very-high-order filter. For example, if the vocal signal were in steady state, it would require one second-order section for each harmonic of the signal. Thus for a low male voice of 40 to 60 harmonics, an order of 80 to 120 would be required. Indeed, our experiments have shown that as the order approaches 100 (or 50 for female voice), the pitch of the vocal sound becomes more and more apparent.

Using the lattice form for the synthesis filter gives us a convenient way of adjusting the order of the filter continuously. Since with the lattice methods the first N sections (coefficients) of the filter are optimal for that order, we can just add one section after another to augment the order. Setting coefficients to zero, starting from the highest, still produces an optimal filter of a lower order.

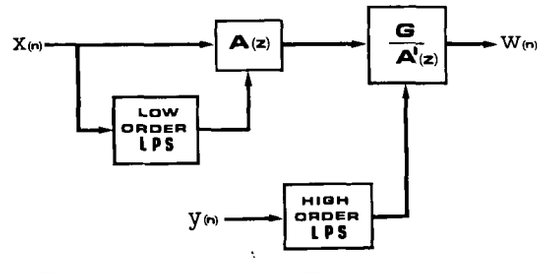

Fig. 6. Diagram of cross synthesis. The source signal $X(n)$ might be a musical instrument. Its spectrum is whitened by a low-order optimum inverse filter, then filtered by a high-order all-pole filter representing the spectrum of another signal $Y(n)$, which is presumably a speech signal of some kind.

This is not true of the direct form or the factored form. Throwing out one coefficient requires changing all the other coefficients to render the filter optimal again. Indeed, instead of just turning a coefficient on or off in the lattice form, we may also turn it up or down. That is to say when we add a new coefficient, we may add it gradually, starting at zero, and slowly advancing to its final value (presumably precomputed). This allows us to "play" the order of the filter, causing the vocal quality to strengthen and fade at will in a continuous manner.

Cross synthesis between musical instruments and voice seems to make the most sense if the two passages are in some way synchronized. We have done this in two different ways to date: one is to record some speech, poetry, or whatever, performed by a professional speaker to achieve the desired presentation, then, using synchronized recording and playing, either through a multitrack tape recorder or through digital recording techniques, have the musician(s) play musical passages exactly synchronized with the vocal sounds. This takes a bit of practice for the musician, in that speech sounds in English are not typically exactly rhythmically precise, but can nonetheless be done quite precisely. The other avenue is to record the music first to achieve some musical performance goal, then have the speaker synchronize the speech with the musical performance. Either one of these approaches achieves synchronicity at the expense of naturalness in one or the other of the performances, vocal or musical, but renders the combination much more convincing.

7. INTERPOLATION

To make the resulting speech as smooth as possible, virtually everything must change smoothly from one point to the next. For instance, if the filter coefficients are changed abruptly at the beginning of a period, there is a perceivable roughness produced. If we wish to interpolate the filter coefficients, however, we must be careful about the choice of a filter structure.

We can envision at least three filter structures: direct form, factored form, and lattice form. In factored form, the filter is realized using first- and second-order sections. The direct form is just a single high-order tapped delay line. The problem with the direct form is that its numerical properties are somewhat less than ideal and that one cannot necessarily interpolate directly the coefficients. If you interpolate linearly between the coefficients of two stable polynomials, the resulting intermediate polynomials are not necessarily stable. Indeed, if the roots of the polynomials are very similar, the intermediate polynomials will probably be stable, but if the roots are very different, the intermediate polynomials are quite likely to be unstable. Thus the direct form is not suitable for interpolation without further thought.

The factored form can be interpolated directly in a stable manner. In a second-order section representing a complex conjugate pole pair, the stability depends largely on the term of delay two. As long as this term is less than unity, the section will probably be stable, depending on the remaining term. There are two problems remaining, though, in the use of the factored form. The first is that the polynomial must be factored. With 55th-order polynomials, this is a nontrivial task. At the time of this writing there is no estimation technique known that can produce the linear prediction filter in already factored form. Although factoring polynomials is an established science, it is still quite time-consuming, especially with high order. In addition to that, one must also group the roots such that each section changes only between roots that are very similar. Since there is no natural ordering of the roots, one must invent a way of so grouping them. We have tried techniques of minimizing the Euclidian distance on the Z plane between pairs of roots, and this seems to give reasonable results, except in certain cases when, for instance, real roots collide and form complex conjugate pairs. There is no telling at any given time how many real roots a polynomial will have, and quite often there are one or two real roots that move around in seemingly random fashion. The factored form does, however, have one strong advantage, which is that it is very clear how to directly modify the spectrum at any given point. Since the roots are already factored, it is quite clear which sections control which parts of the spectrum. Moreover, when the roots are interpolated, they form clear, well-defined patterns that have well-defined effects on the spectrum. Except for the inefficiencies involved in factoring and ordering the roots, the factored form seems ideal.

With the lattice form there is no problem in interpolation. The reflection coefficients can be interpolated directly without fear of instabilities because the condition for stability is simply that each coefficient be of magnitude less than unity. When one interpolates, however, between reflection coefficients of two stable filters, the roots follow very complex paths, thus if the filters are not already very similar, one can only expect that the intermediate filters will be very loosely related to the original filters.

If one wishes to modify the spectrum, however, one must convert the reflection coefficients into direct form and then factor the polynomial. This can be done without problem with the expenditure of sufficient quantities of computer time, but the inverse process, converting the direct form back into reflection coefficients, cannot be done accurately. The only process for doing so is highly numerically unstable [7], so that for higher orders it simply cannot be done in reasonable amounts of time. Thus once factored, the polynomial must stay factored for all time thereafter.

For our own synthesis system, we currently use the lattice form because it uses directly the output of the analysis technique and because interpolation can be used easily on the reflection coefficients.

Filter coefficients are, of course, not the only things that must be interpolated. The frequency must also be continuously interpolated for the most smooth sounding results. This is where the advantage of using table lookup for the excitation occurs. With table lookup one can continuously vary the rate at which the table is scanned. If one uses interpolation on the table itself, the resulting process can be made very smooth indeed. Again, the table must be regenerated each time the frequency changes significantly,

but this seems to occur seldom enough to allow the usage of the FFT for generating the excitation function.

8. AMPLITUDE CONTROL

The amplitude of the synthetic signal should be controlled to produce a loudness contour that corresponds as much as possible to the original loudness. As mentioned by Moorer [26], what would be ideal is some kind of direct loudness normalization, using possibly a model of human loudness perception [29]. Unfortunately this computation is so unwieldy as to render it virtually useless at this time, so some other methods must be chosen.

Atal and Hanauer [4] used a method of normalization of energy such that the energy of the current frame (period) is scaled to correspond exactly to the original energy. Although this sounds like the right thing to do, it has several problems. The first is just the way it is calculated. The filter has presumably been run on the previous frame and now has a nonzero "memory." That means that even with zero input, this frame will emit a certain response that will presumably die away. We seek, then, to scale the excitation for this frame such that the combination of the remaining response from the previous frame and the response for this frame (starting with a fresh filter for this frame) will have the correct energy. Since the criterion is energy, a squared value, this reduces to the solution of a quadratic equation for the gain factor. The problem comes when the energy represented by the tail of the filter response from the previous frame already exceeds the desired energy of this frame. In this case the solution of the quadratic is, of course, complex. What this means is that the model being used is imperfect. Either the filter or the excitation is not an accurate model of the input signal. This is possible since the modeling process, especially for the excitation, is not an exact procedure. There are even instabilities that can result in the computation of the gain. For instance, if the response from the last frame is large, but not quite as large as the desired energy, then a very small value of gain will be computed. That means that in the next frame there will be very little contribution from the previous frame, and the gain factor will be quite large. As the model deteriorates, this oscillation in the gain increases until no solution is possible. The only hope is that this occurs sufficiently rarely as to not be a detriment. Experience, however, seems to indicate the contrary: this failure in modeling is something that happens even in quite normal speech and must be taken into account. Besides all that, even if you do normalize the energy, the perceived loudness will often be found to change noticeably over the course of the utterance. This is especially true during voiced fricatives, although the theoretical explanation for this phenomenon is not clear at this time.

The method of amplitude control that we have chosen is twofold: for cross synthesis we choose a two-pass post-normalization scheme that computes the energies of the original signal and the synthesized signal. The synthesized signal is then multiplied by a piecewise-linear function, the breakpoints of which are the gain factors required to normalize the energy at the points where the energies were computed. For resynthesis of the vocal sounds we use an open-loop method of just driving the filter with an excitation that corresponds in energy to the energy of the error signal of the inverse filter. This is, of course, only an approximation because the excitation never corresponds to the actual error signal, but in practice it seems to produce the smoothest most naturally varying sounds. Note also that this does not guarantee any correspondence between the energies of the original and the synthetic signals. With the autocorrelation method of linear prediction, the error energy is easily obtained as an automatic result of the filter computation. For other methods it is generally necessary to actually apply the filter to the original signal to obtain the error energy. As with all other parameters, we interpolate the gain in a continuous piecewise-linear manner throughout the synthesis.

9. WHERE TO FROM HERE?

Problems remain in certain areas, such as the synthesis of nasal consonants and the voiced–unvoiced–silence decision. With nasal consonants it is theorized that the presence of the nasal zero must be simulated in the filter. This cannot be entirely true because some nasals can be synthesized quite well and some cannot. Additional work must be done to try to distinguish the features of the nasals that do not adapt well to the linear prediction method and decide what is to be done about them. Some amount of work has been done on the simultaneous estimation of poles and zeros [30], [31], and we will be very interested to examine the results in critical listening tests. The voiced–unvoiced–silence decision may well require hand correction for the forseeable future.

These techniques have been embodied in a series of programs that allow the composer to specify transformations on the timing, pitch, and other parameters in terms of a piecewise-linear functions that can be defined directly in terms of their breakpoints, graphically, or implicitly in terms of resulting contours of time, pitch, or whatever. More work must be done in arranging these in a more convenient package for smoothly carrying the system through from start to finish without excessive juggling and hit-or-miss estimation.

10. REFERENCES

[1] J. P. Burg, "Maximum Entropy Spectral Analysis," presented at the 37th Annual Meeting of the Soc. Explor. Geophy., Oklahoma City, OK 1967.

[2] E. A. Robinson, *Statistical Communication and Detection* (Hafner, New York, 1967).

[3] B. S. Atal and M. R. Schroeder, "Adaptive Predictive Coding of Speech Signals," *Bell Sys. Tech. J.*, vol. 49, pp. 1973–1986 (1970).

[4] B. S. Atal and S. L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," *J. Acoust. Soc. Am.*, vol. 50, pp. 637–655 (1971 Feb.).

[5] F. Itakura and S. Saito, "Digital Filtering Techniques for Speech Analysis and Synthesis," presented at the 7th International Congress on Acoustics, Budapest, 1971, paper 25-C-1.

[6] J. Makhoul, "Linear Prediction: A Tutorial Re-

view," *Proc. IEEE,* vol. 63, pp. 561–580 (1975 Apr.).

[7] J. D. Markel and A. H. Gray, *Linear Prediction of Speech* (Springer, Berlin-Heidelberg, 1976).

[8] C. Dodge, "Synthetic Speech Music," (disk), Composer's Recordings, New York, CRI-SD-348 (1975).

[9] J. A. Moorer, "Signal Processing Aspects of Computer Music: A Survey," *Proc. IEEE,* vol. 65, pp. 1108–1137 (1977 Aug.).

[10] T. L. Petersen, "Vocal Tract Modulation of Instrumental Sounds by Digital Filtering," presented at the Music Computation Conf. II, School of Music, University of Illinois, Urbana-Champaign, 1975 Nov. 7–9.

[11] T. L. Petersen, "Dynamic Sound Processing," *Proc. 1976 ACM Computer Science Conf.,* Anaheim, CA, 1976 Feb. 10–12.

[12] T. L. Petersen, "Analysis-Synthesis as a Tool for Creating New Families of Sound," presented at the 54th Convention of the Audio Engineering Society, Los Angeles, 1976 May 4–7.

[13] J. Makhoul, "Lattice Methods for Linear Prediction," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, (1977 Oct.).

[14] A. M. Noll, "Cepstrum Pitch Determination," *J. Acoust. Soc. Am.,* vol. 41, pp. 293–309 (1967 Feb.).

[15] B. Gold and L. R. Rabiner, "Parallel Processing Techniques for Estimating Pitch Periods of Speech in the Time Domain," *J. Acoust. Soc. Am.,* vol. 46, pp. 442–448 (1969 Aug.).

[16] M. M. Sondhi, "New Methods of Pitch Extraction," *IEEE Trans. Audio Electroacoust.,* vol. AU-16, pp. 262–266 (1968 June).

[17] J. A. Moorer, "The Optimum Comb Method of Pitch Period Analysis of Continuous Digitized Speech," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-22, pp. 330–338 (1974 Oct.).

[18] B. S. Atal and L. R. Rabiner, "A Pattern Recognition Approach to Voiced–Unvoiced–Silence Classification with Applications to Speech Recognition," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-24, pp. 201–211 (1976 June).

[19] L. R. Rabiner and M. R. Sambur, "Application of an LPC Distance Measure to the Voiced–Unvoiced–Silence Detection Problem," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, pp. 338–343 (1977 Aug.).

[20] L. R. Rabiner, B. S. Atal, and M. R. Sambur, "LPC Prediction Error—Analysis of Its Variation with the Position of the Analysis Frame," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, pp. 434–442 (1977 Oct.).

[21] D. R. Morgan and S. E. Craig, "Real-Time Adaptive Linear Prediction Using the Least Mean Square Gradient Algorithm," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-24, pp. 494–507 (1976 Dec.).

[22] M. Morf, "Fast Algorithms for Multivariable Systems," PhD thesis, Dept. Elec. Eng., Stanford University, Stanford, CA, 1974.

[23] M. Morf, A. Vieira, D. T. Lee, and T. Kailath, "Recursive Multichannel Maximum Entropy Method," *Proc. 1977 Joint Automatic Control Conf.,* San Francisco, CA, 1977.

[24] G. Winham and K. Steiglitz, "Input Generators for Digital Sound Synthesis," *J. Acoust. Soc. Am.,* vol. 47, pp. 665–666 (1970).

[25] D. E. Knuth, *The Art of Computer Programming,* vol. 2: *Seminumerical Algorithms,* (Addison-Wesley, Reading, MA, 1969.

[26] J. A. Moorer, "The Synthesis of Complex Audio Spectra by Means of Discrete Summation Formulas," *J. Audio Eng. Soc.,* vol. 24, pp. 717–727 (1976 Nov.).

[27] M. R. Portnoff, "Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-24, pp. 243–248 (1976 June).

[28] J. B. Allen, "Short-Term Spectral Analysis, Synthesis, and Modification by Discrete Fourier Transform," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, pp. 235–238 (1977 June).

[29] E. Zwicker and B. Scharf, "A Model of Loudness Summation," *Psychol. Rev.,* vol. 1, pp. 3–26 (1965).

[30] K. Steiglitz, "On the Simultaneous Estimation of Poles and Zeros in Speech Analysis," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, pp. 229–234 (1977 June).

[31] J. M. Tribolet, "Identification of Linear Discrete Systems with Applications to Speech Processing," MS thesis, Dept. of Elec. Eng., M.I.T., Cambridge, MA, 1974 Jan.

[32] C. A. McGonegal, L. R. Rabiner and A. E. Rosenberg, "A Subjective Evaluation of Pitch Detection Methods Using LPC Synthesized Speech," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-25, pp. 221–229 (1977 June).

[33] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal, "A Comparative Performance Study of Several Pitch Detection Algorithms," *IEEE Trans. Acoust., Speech, Signal Proc.,* vol. ASSP-24, pp. 399–418 (1976 Oct.).

THE AUTHOR

James Anderson Moorer was born November 25, 1945 in Hollywood, Florida. He received his SB degree in electrical engineering from the Massachusetts Institute of Technology in 1967 and an SB degree in applied mathematics in 1968. Dr. Moorer later graduated from Stanford University with a PhD in computer science in 1975.

He worked for four years as systems programmer at the Stanford Artificial Intelligence Laboratory. He is a Research Associate with the newly-formed Stanford Center for Computer Research in Music and Acoustics. Currently on leave from Stanford, he is at the Institute de Recherche et Coordination Acoustique/Musique at the Pompidou Center in Paris.