Biol. Cybern. 60, 267-276 (1989)

Biological Cybernetics

A Robust Orthogonal Algorithm for System Identification and Time-Series Analysis

M. J. Korenberg

Department of Electrical Engineering, Queen's University, Kingston, Ontario, Canada K7L 3N6

Abstract. We describe and illustrate methods for obtaining a parsimonious sinusoidal series representation or model of biological time-series data. The methods are also used to identify nonlinear systems with unknown structure. A key aspect is a rapid search for significant terms to include in the model for the system or the time-series. For example, the methods use fast and robust orthogonal searches for significant frequencies in the time-series, and differ from conventional Fourier series analysis in several important respects. In particular, the frequencies in our resulting sinusoidal series need not be commensurate, nor integral multiples of the fundamental frequency corresponding to the record length. Freed of these restrictions, the methods produce a more economical sinusoidal series representation (than a Fourier series), finding the most significant frequencies first, and automatically determine model order. The methods are also capable of higher resolution than a conventional Fourier series analysis. In addition, the methods can cope with unequally-spaced or missing data, and are applicable to time-series corrupted by noise. Finally, we compare one of our methods with a well-known technique for resolving sinusoidal signals in noise using published data for the test time-series.

1 Introduction

The estimation of power spectral density for time-series analysis and the resolution of two or more sinusoidal signals in the presence of noise are important problems in biology. Here we describe and illustrate new procedures (Korenberg 1987, 1988) for systematically constructing a parsimonious sinusoidal series representation or model of biological time-series data. The procedures are also applied below to identify nonlinear systems of unknown structure.

Since the rediscovery of the fast Fourier transform (FFT) by Cooley and Tukey (1965), there has been great interest in the computation of power spectra of time-series data (Box and Jenkins 1976). Much use has been made of spectral analysis techniques in the life sciences. For example, the techniques are frequently applied to analyse electroencephalograms (EEG), e.g. in epileptics (Sterman 1981). The FFT is often used to obtain power spectra of respiration and heart rate, e.g. in newborn or young infants (Nugent and Finley 1983). However the FFT approach often works poorly for short data records (Kay and Marple 1981; Mohanty 1986). Moreover, as is frequently pointed out (e.g. Marmarelis and Marmarelis 1978), Fourier transforming a signal is a numerically ill-posed problem. That is, small amounts of noise corruption of the time-series data can cause large errors in the frequency domain. For this reason, various smoothing procedures have been developed which involve the use of spectral windows such as the Hamming, Hanning and Parzen windows (Marmarelis and Marmarelis 1978; Mohanty 1986). These procedures reduce statistical error at the expense of resolution, resulting in leakage or smearing (Marmarelis and Marmarelis 1978) in the Fourier series representation.

Consequently, various authors have investigated the estimation of power spectral density using parametric methods (Mohanty 1986), e.g. autoregressive moving average (ARMA), maximum entropy method, and maximum likelihood estimator. In the first of these approaches, the estimated spectrum is readily computed once the parameters in an ARMA or an autoregressive (AR) model have been identified. However, this identification must be carried out with only the time-series data available, an "inaccessible input" problem. Various procedures for doing this have been developed (Box and Jenkins 1976). However, noise corruption of the time-series data can seriously affect the resolution of some of these procedures.

The Prony and Pisarenko methods (Mohanty 1986) can be used to estimate a sum of sinusoidal signals in the presence of noise. Mohanty (1986), following Kay and Marple (1981), notes that the latter two methods offer high resolution, but model order must be selected (as also required by the AR and maximum entropy methods), and a polynomial equation must be solved.

In the present paper two very simple approaches, fast orthogonal search (Korenberg 1987, 1988) and robust orthogonal search, are described (and the fast search is illustrated) for finding the significant frequencies in time-series data. Both approaches are effective with short data records, and cope with noisy, missing and unequally-spaced data. For these reasons, the methods appear suitable for analysis of biological data, e.g. EEG, electromyogram (EMG), electrooculogram (EOG), electrocardiogram (ECG) and beat-to-beat heart rate time-series. In particular, the methods enable a parsimonious sinusoidal series representation or model to be systematically developed for the time-series. One reason for the parsimony of the representation is that the component frequencies need not be commensurate nor integral multiples of a fundamental frequency. This is an important difference from a conventional Fourier series, where all frequencies must be harmonics of a fundamental determined by the record length. Yet the significant frequencies in a given time-series may be unrelated to the record length. Hence conventional Fourier series analysis seeks to develop a sinusoidal series representation based on a parameter (record length) which might not be effective in producing an economical representation.

In both fast and robust orthogonal searches, the most significant sinusoidal frequencies are found first. Moreover simple techniques for measuring mean-square error and its potential reduction at any stage are used in determining when to stop further development of the representation, thereby automatically establishing model order. The methods presented are also capable of higher resolution than obtainable by conventional Fourier series analysis. In simulations below, we compare the performance of fast orthogonal search with the Prony method for resolving a sum of sinusoidal signals embedded in noise.

The application of fast orthogonal search (Korenberg 1987, 1988) and robust orthogonal search to time-series analysis can best be understood by first considering how they are applied to system identification. For the identification of nonlinear systems, Wiener (1958) pioneered the use of the Gram-Schmidt process on a functional expansion to produce an orthogonal series. Wiener kernels in his orthogonal series were determinable in the time domain by the Lee-Schetzen (1965) crosscorrelation formula or via the frequency domain by the method of French and Butz (1973). Important applications, particularly in neurophysiology (Marmarelis and Marmarelis 1978; Marmarelis and Naka 1972), and extensions (Palm and Poggio 1978) of Wiener's orthogonal approach have followed. Such work has indeed motivated the present paper.

2 Fast Orthogonal Search and System Identification

2.1 System Model

Consider approximating a nonlinear system by the difference equation model (Haber and Keviczky 1976; Billings and Leontaritis 1982)

$$y(n)=F[y(n-1),\ldots,y(n-K),x(n),\ldots,x(n-L)]+e(n). \tag{1}$$

Here F is a polynomial, $x(n)$ is the system input, $y(n)$ is the system output, and $e(n)$ is the equation error. Assume the data record is defined for $n=0,\ldots,N$. Equation (1) can be expressed more concisely as

$$y(n)=\sum_{m=0}^{M} a_m p_m(n)+e(n). \tag{2}$$

Here the a_m are the difference equation coefficients,

$$p_0(n)=1 \tag{3}$$

and the remaining $p_m(n)$ are chosen from the x and y terms on the right side of (1) and cross-products thereof (including powers):

$$p_m(n)=y(n-k_1)\ldots y(n-k_i)x(n-l_1)\ldots x(n-l_j) \tag{4}$$

$$m\geqq 1,\quad i\geqq 0,\quad 1\leqq k_1\leqq K,\ldots,1\leqq k_i\leqq K$$

$$j\geqq 0,\quad 0\leqq l_1\leqq L,\ldots,0\leqq l_j\leqq L.$$

The crucial selection of terms is now discussed briefly.

2.2 Selecting Model Terms by Orthogonal Search Method

An orthogonal search method was developed (Korenberg 1985; McIlroy 1986) for efficiently obtaining difference equation models of nonlinear systems with unknown structure. Using this method, we can build up an economical series representation

$$y(n)=\sum_{m=0}^{M} g_m w_m(n)+e(n), \tag{5}$$

where the $w_m(n)$ are orthogonal over the data record, and the g_m are the orthogonal expansion coefficients (achieving a least-squares fit). Central to the orthogonal search method is first to use Gram-Schmidt orthogonalization to construct, from each candidate term, a function which is orthogonal to all previously

chosen terms. Then the reduction in mean-square error achievable by selecting any given candidate is readily obtained from the norm of the corresponding orthogonal function and the orthogonal expansion coefficient. More precisely, suppose a further term $a_{M+1}p_{M+1}(n)$ is to be added to the right side of the model in (2). Then the mean-square error (of the model fit) will be reduced by the amount

$$Q(M+1)=g_{M+1}^2\overline{w_{M+1}^2(n)}. \tag{6}$$

Here the overbar denotes the time-average from $n=0$ to $n=N$.

To expand the difference equation model by a further term using the orthogonal search method (Korenberg 1985; McIlroy 1986), evaluate the quantity Q in (6) for each candidate addition. Choose the candidate for which Q is greatest, since this addition will result in the greatest reduction in mean-square error. By continuing in this way, it is possible to efficiently construct accurate parsimonious models of real systems, particularly if a threshold level is used to reject unsuitable terms (McIlroy 1986).

However, the creation of the orthogonal functions $w_m(n)$ in (5) is expensive in computing time and memory, and is avoided if model terms are selected by the fast orthogonal search.

2.3 Selecting Model Terms by Fast Orthogonal Search

Fast orthogonal search was developed (Korenberg 1987, 1988) as a more efficient method of building models of time-series and of systems with unknown structure. Only the essential details for present purposes are summarized here. The method uses a Cholesky decomposition as follows. The orthogonal expansion coefficients in (5) are given by

$$g_m=\frac{C(m)}{D(m,m)} \qquad m=0,\ldots,M, \tag{7}$$

where

$$D(0,0)=1, \tag{8}$$

$$D(m,0)=\overline{p_m(n)}, \qquad m=1,\ldots,M, \tag{9}$$

$$D(m,r)=\overline{p_m(n)p_r(n)}-\sum_{i=0}^{r-1}\alpha_{ri}D(m,i) \qquad m=1,\ldots,M;\ r=1,\ldots,m, \tag{10}$$

$$\alpha_{mr}=\frac{D(m,r)}{D(r,r)} \qquad m=1,\ldots,M;\ r=0,\ldots,m-1, \tag{11}$$

$$C(0)=\overline{y(n)}, \tag{12}$$

$$C(m)=\overline{y(n)p_m(n)}-\sum_{r=0}^{m-1}\alpha_{mr}C(r) \qquad m=1,\ldots,M. \tag{13}$$

If the g_m and α_{mr} were known, the coefficients a_m in (2) could be obtained by the following formula (Korenberg et al. 1988a):

$$a_m=\sum_{i=m}^{M}g_iv_i, \tag{14A}$$

where

$$v_m=1, \tag{14B}$$

$$v_i=-\sum_{r=m}^{i-1}\alpha_{ir}v_r, \qquad i=m+1,\ldots,M. \tag{14C}$$

Suppose that the model terms $p_m(n)$ had already been selected. Then, the α_{mr} could be calculated from the following pseudocode, which achieves a Cholesky factorization:

```
D(0,0)=1
FOR m=1 TO M
Calculate D(m,0) from (9)
NEXT m
FOR m=1 TO M
FOR r=0 TO m-1
Calculate α_mr from (11)
Calculate D(m,r+1) using (10)
NEXT r
NEXT m.
```

Then $C(m)$ could be solved for recursively from (12) and (13), $m=0,\ldots,M$, and the g_m follow from (7). Finally, the required coefficients a_m in (2) could be obtained from (14).

However in practice it is efficient to build up the model by selecting one further term at a time, and we achieve this by the following departure from the above. It can be shown (Korenberg 1987) that the mean-square error (of the model fit)

$$\text{m.s.e.}=\overline{\left(y(n)-\sum_{m=0}^{M}a_mp_m(n)\right)^2} \tag{15}$$

is equivalently

$$\text{m.s.e.}=\overline{y^2(n)}-\sum_{m=0}^{M}g_m^2D(m,m). \tag{16}$$

Suppose that $a_Mp_M(n)$ was the last term added to the model of (2). Then the addition of this term reduced the mean-square error by the amount

$$Q(M)=g_M^2D(M,M). \tag{17}$$

Equation (17) makes it possible to assess rapidly the benefit from adding any given candidate term to

the model, without requiring the creation of the orthogonal functions $w_m(n)$. Consider screening the candidates for $p_M(n)$ where $M \geqq 1$. For each such candidate, use the equations in this subsection to calculate the corresponding g_M and $D(M, M)$, and thus measure the effectiveness of the candidate using (17). The candidate with largest Q value is selected (optionally, subject to exceeding a specified positive threshold level). [It is also useful to ensure that $D(M, M)$ exceeds a specified positive threshold level. Avoiding candidates unable to meet the latter threshold requirement prevents division by neglegibly small numbers in (7) and (11), with resulting inaccuracies.] Suppose that we have just chosen $p_M(n)$. If the mean-square error given by (16) is not acceptably small, then similarly search the remaining candidate terms to determine the choice for $p_{M+1}(n)$, and so on. In this way, we can rapidly construct an accurate, parsimonious model for a system of unknown structure.

Consider again the search for $p_M(n)$. In obtaining $D(M, M)$ and g_M for a given candidate, we can avoid repeating calculations done previously. For example, the above pseudocode merely need be carried out with $m = M$, and not for earlier values of m. Similarly, it suffices to set $m = M$ in (13) to obtain $C(M)$ [and hence g_M from (7)]. Once the choice for $p_M(n)$ has been made, repeat the abbreviated pseudocode for the chosen candidate, and similarly recalculate $C(M)$ from (13) and g_M from (7). This is done to properly set the values of the α_{Mr}, $D(M, M)$, $C(M)$ and g_M prior to searching for $p_{M+1}(n)$ (assuming the mean-square error given by (16) is not acceptably small).

Also note that using (9), (10), (12), and (13) requires knowledge of the time-averages appearing on the right sides of these equations. Clearly the time-average on the right side of (12) need be computed once only at the outset and stored. The time-averages on the right sides of (9), (13), and (10) when $r = m$, can also be computed once and for all at the outset [letting $p_m(n)$ in turn equal each candidate term] and stored. Then, in order to be able to test a given candidate term at any stage, merely compute and store the time-average of that candidate with the last *chosen* term. All other required time-averages will be available from the previous rounds of searching.

Finally, note that the lagged construction of the difference equation terms $p_m(n)$ in (4) makes it possible to evaluate efficiently the time-averages on the right sides of (9) and (10) (Korenberg 1988). This is done by relating the time-averages to input and/or output means and correlations and making small corrections for the finite data record. Such an approach does not require explicit creation of the terms $p_m(n)$, and moreover is much less time-consuming than calculating each time-average independently.

3 Fast Orthogonal Search and Time-Series Analysis

3.1 Approximation by a Sum of Exponentials

Both fast and robust orthogonal searches can be used directly to fit a given time-series by a sum of exponentials (for example, when the time-series represents samples from the complimentary solution of a linear differential equation). Thus the time-series data is treated as the system output, while the candidate terms for approximating the output are the exponential functions. In more detail, (3) still holds, and for $m \geqq 1$

$$p_m(n) = \exp(-n/\beta_m). \tag{18}$$

For $m = 1, 2, \ldots$, the β_m are successively chosen by searching through a set of candidate "time-constants", analogously as set out in Sect. 2.3. Finally, (14) can be used to obtain the coefficients a_m of the exponential series which (2) will represent.

The fast and robust orthogonal searches can similarly be used to fit time-series with exponentially-decaying sinusoids, logarithmic functions, splines, polynomials, Walsh and gate functions, and a wide variety of other functions. Of special interest are periodic functions such as triangle and square-waves, and trigonometric and hypergeometric functions. Below we concentrate on obtaining parsimonious sinusoidal series representations of time-series data (Korenberg 1987).

3.2 Approximation by a Sinusoidal Series

In what follows, (2) represents a sinusoidal series representation which is systematically constructed for the time-series data $y(n)$, $n = 0, \ldots, N$. (The number of data points need not be a power of two, and padding of the time-series is unnecessary.) Equation (3) still holds and for $i = 1, 2, \ldots$

$$p_{2i-1}(n) = \cos\omega_i n, \tag{19}$$

$$p_{2i}(n) = \sin\omega_i n. \tag{20}$$

For simplicity, we assume initially that the time-series data is equally spaced. (Below we will consider a simple modification to cope with missing or unequally-spaced data.)

The ω_i are determined successively by searching through a set of candidate frequencies $\omega_A, \omega_B, \ldots$ which need not be commensurate. It can be shown (Korenberg 1987) that adding the i-th term pair

$$T_i = a_{2i-1}p_{2i-1}(n) + a_{2i}p_{2i}(n) \tag{21}$$

to the sinusoidal series model reduces the mean-square error by the amount

$$Q_i(i) = g_{2i-1}^2 D(2i-1, 2i-1) + g_{2i}^2 D(2i, 2i). \tag{22}$$

In (21), a_{2i-1} and a_{2i} are respectively the cosine and sine amplitudes. At the stage of adding the term pair in (21), $M=2i$.

Thus, begin by introducing a constant into the sinusoidal series using (3), and obtain g_0 from (7), (8), (12). Then, to find ω_1, for each candidate frequency ω_A, ω_B, ... evaluate $Q_1(1)$ using (22). Finally, set ω_1 equal to the candidate frequency with largest Q_1 value (optionally, subject to exceeding a threshold level). If the mean-square error given by (16) (with $M=2$) is not an acceptably small percentage of the variance of the time-series, then search for a value for ω_2, and so on. In general to find ω_i, $i=1,2,\ldots$, for each candidate frequency not previously selected evaluate $Q_1(i)$. Choose the candidate for which $Q_1(i)$ is largest (again, optionally, subject to exceeding a threshold level). (Also, we again wish to ensure that, for each model term, $D(m,m)$ is not neglegibly small.) Continue the process unless the mean-square error given by (16) (with $M=2i$) is acceptably small, or the model has reached the maximum size allowable. The process may also be terminated if no remaining candidate frequency can cause a reduction in mean-square error exceeding a specified threshold value. At this point (14) can be used to find the coefficients a_m in the sinusoidal series model represented by (2). The identified model can then provide a synthesized approximation to the original time-series. Since this approximation is constructed with only the most significant frequencies in the time-series, it tends to be far less noisy than the original data (see example below).

Recall that at the stage of adding the i-th term pair, $M=2i$. To evaluate $Q_1(i)$ for a given candidate requires knowing the corresponding g_m, $D(m,m)$, when $m=2i-1$ and $m=2i$. To avoid repeating calculations already performed, (13) and the pseudocode in Sect. 2.3 need be carried out merely for these two values of m. In particular we can use the following abbreviated pseudocode:

$D(0,0)=1$

FOR $m=M-1$ TO M

Calculate $D(m,0)$ from (9)

NEXT m

FOR $m=M-1$ TO M

FOR $r=0$ TO $m-1$

Calculate α_{mr} from (11)

Calculate $D(m,r+1)$ using (10)

NEXT r

NEXT m.

Suppose the choice for ω_i has just been determined. Then carry out the abbreviated pseudocode immediately above, and (13) with $m=M-1$, M, using the chosen candidate frequency to define the $p_{2i-1}(n)$ and $p_{2i}(n)$ in (19), (20). This is done to properly set the values of the α_{mr}, $D(m,m)$, $C(m)$ and g_m, $m=2i-1$, $2i$ ($r=0,\ldots,m-1$), prior to either searching for ω_{i+1} or terminating the process and using (14) to calculate the final model of (2).

Use of (9), (10), (12), (13) requires knowing the time-averages appearing on the right sides of these equations. The time-average on the right side of (12) (i.e. the average of the time-series) need be computed only once and then stored. Similarly, the time-average on the right sides of (9), (13), and (10) when $r=m$, can be calculated (and stored) once and for all at the outset, letting $p_m(n)$ equal in turn the sine and cosine function corresponding to each candidate frequency. Then before using (9), (10), (13) in testing a new candidate frequency, one merely computes the time-average of the four sine and cosine pairs involving the candidate frequency and the last chosen frequency. Thus suppose ω_i was the last sinusoidal frequency determined, and we wish to test a given candidate (say ω_A) for ω_{i+1}. Then we first measure

$$\overline{\sin\omega_A n \sin\omega_i n}, \quad \overline{\cos\omega_A n \sin\omega_i n}, \quad \overline{\sin\omega_A n \cos\omega_i n}, \quad \overline{\cos\omega_A n \cos\omega_i n}. \tag{23}$$

All other required time-averages will be available from the previous rounds of searching. Moreover, when the data is equally-spaced, the time-averages on the right-sides of (9) and (10) are readily obtainable (Korenberg 1987) using well-known formulas (Dwight 1960).

Further details are available (Korenberg 1987) concerning the fast orthogonal search and the application of this method and the orthogonal search method to time-series analysis. Related techniques of the present author have been applied to real time-series data in projects supervised by him in the Department of Electrical Engineering at Queen's University, e.g. Ho et al. (1987). In the latter study, to cope with unequally-spaced data, the variable n on the right sides of (19), (20) was replaced by the function $t(n)$, which corresponded to the actual timing of the n-th data sample. The same strategy can be used to cope with unequally-spaced or missing data in the fast and robust orthogonal searches described in the present paper.

4 Example

The test time-series was listed in a study of various spectral estimation techniques by Kay and Marple (1981), and appears again in the review by Mohanty

(1986). The 64-point real sample sequence was generated by adding a passband noise process to a sum of sinusoids of fractional frequencies 0.1, 0.2, and 0.21 and amplitudes 0.1, 1, and 1 respectively. (Frequencies are expressed as a fraction of the sampling frequency.) The noise process had been colored by filtering white Gaussian noise, and was centered at a frequency of 0.35. Both Kay and Marple (1981) and Mohanty (1986) show the true power spectral density for the time-series, and those estimated by various techniques including the Prony method and its spectral line variant.

As Kay and Marple (1981) point out, the sinusoidal frequencies 0.2 and 0.21 are closer than the resolution width of the FFT, whose nominal resolution in Hz is 1/64 of the sampling frequency. Indeed they found that the FFT did not resolve these two frequencies.

Two tests of fast orthogonal search were conducted. We first tested its ability to model both the narrow-band and wide-band processes. Second, we tested its ability to recover an accurate estimate of the time-series prior to its corruption by the passband noise process. In both tests, 100 candidate frequencies, equally-spaced between 0 and 0.5 times the sampling frequency, were searched and up to 20 distinct frequencies were permitted in the final model. Note that fast orthogonal search does not require the number of data points to be a power of two, as happened here.

4.1 Modelling Narrow- and Wide-Band Processes

We first tested the capability of fast orthogonal search to achieve accurate spectral estimation of narrow- and wide-band processes. To do this, we set a low threshold (0.2% of the time-series variance) as the minimum reduction in mean square error required before a further addition could be made to the model. Fast orthogonal search estimated a constant (-2.86×10^{-3}) and selected 12 sinusoidal frequencies, resulting in a mean-square error equal to 0.483% of the time-series variance. Frequency components were selected in the following order (top number is fractional frequency, middle and bottom numbers are sine and cosine amplitude respectively):

0.210	0.200	0.365	0.310	0.400	0.350
2.859×10^{-2}	-1.829×10^{-3}	0.172	0.139	0.198	2.614×10^{-2}
0.990	0.971	−0.191	−0.186	-6.653×10^{-3}	−0.167
0.295	0.415	0.330	0.100	0.270	0.435
0.122	-9.805×10^{-3}	-3.289×10^{-2}	-1.162×10^{-3}	5.617×10^{-2}	1.236×10^{-2}
8.175×10^{-2}	0.113	0.101	9.948×10^{-2}	-5.372×10^{-2}	-7.349×10^{-2}

For example, fast orthogonal search selected a first frequency equal to 0.210, with a sine amplitude of 2.859×10^{-2} and a cosine amplitude of 0.990. Notice fast orthogonal search was able to resolve the fractional frequencies 0.2 and 0.21, which the FFT was unable to do (Kay and Marple 1981). The amplitude of the sinusoidal components at fractional frequencies 0.1, 0.2, 0.21 were estimated as 0.099, 0.971, 0.990 by fast orthogonal search (actual values were 0.1, 1, 1). The corresponding Prony method estimates (Kay and Marple 1981) of the fractional frequencies (0.100, 0.201, 0.209) and respective amplitudes (0.092, 1.276, 0.845) were less accurate. Prony spectral line method estimates (Kay and Marple 1981) of the fractional frequencies 0.1, 0.2, 0.21 and corresponding amplitudes were comparable to, but phase estimates were less accurate than, fast orthogonal search estimates. Fast orthogonal search also modelled the broad-band process more accurately than either variant of the Prony method. Each of the Prony variants selected 8 sinusoidal frequencies in total to model the narrow- and broad-band processes.

4.2 Estimating the Noise-Free Time-Series

Here we attempt to recover the time-series as it existed prior to corruption by the noise process. To do this, we again applied fast orthogonal search to the test time-series (hereafter called "noisy"). However, we used a higher threshold (4% of the noisy time-series variance) as the minimum reduction in mean-square error required before a further addition could be made to the model. Fast orthogonal search estimated a constant (3.568×10^{-3}) and selected two fractional frequencies, resulting in a mean-square error of 13.39% (of the noisy time-series variance). In order, the following frequencies, sine and cosine amplitudes were selected (set out in the same style as before):

0.210	0.200
2.983×10^{-2}	2.760×10^{-2}
1.027	0.997

(24)

Using the estimated constant, and the sinusoidal components identified in (24), we synthesized our

estimate $z(n)$ of the noise-free time-series $Z(n)$. We defined the noise-free time-series by

$$Z(n)=0.1\cos(0.2\pi n)+\cos(0.4\pi n)+\cos(0.42\pi n) \tag{25}$$

and computed the signal-to-noise ratio

$$\text{SNR}=\frac{\text{variance }[Z(n)]}{\text{variance }[z(n)-Z(n)]}. \tag{26}$$

The synthesized estimate $z(n)$ of the noise-free time-series resulted in a SNR = 142.81 (21.55 dB). This should be compared with the SNR of 6.63 (8.21 dB) when the noisy time-series was used in place of $z(n)$ in (26). Thus fast orthogonal search can be used to reduce dramatically the degree of noise corruption.

In summary, suppose it is desired to obtain accurate spectral estimation of a signal via fast orthogonal search. Then we use a low threshold for the minimum reduction in mean-square error required before a further addition can be made to the model. This results in a final model with a low mean-square error. Suppose instead we wish to synthesize an approximation of the noise-free time series from noisy data. Then a higher threshold is selected, and the final model will have a larger mean-square error when compared with the noisy time-series. This of course is desirable if one is to avoid fitting the noise process corrupting the time-series.

5 Robust Orthogonal Search

Our pseudocode in Sects. 2.3 and 3.2 for carrying out the Cholesky factorization is ultimately founded on Gram-Schmidt orthogonalization. The Gram-Schmidt process is known to be less numerically-stable than the modified Gram-Schmidt procedure (Rice 1966). Thus, by adapting the equations for carrying out the modified Gram-Schmidt procedure, we can produce a more robust method for achieving a Cholesky factorization.

Suppose again that the $p_m(n)$, $m=0,\ldots,M$, form a set of terms from which the orthogonal functions are to be constructed. Under the modified Gram-Schmidt procedure, one term is taken to be the first orthogonal function and weighted amounts of this term are subtracted from all remaining terms to form a new set. One term is selected from the new set to be the second orthogonal function, weighted amounts of this term are subtracted from all remaining terms in the new set, and so on. Let $p_0(n)$ again be defined by (3), and let

$$p_m^{(0)}(n)=p_m(n), \quad m=0,\ldots,M. \tag{27}$$

For $j=0,\ldots,M-1$ and $m=j+1,\ldots,M$, let

$$p_m^{(j+1)}(n)=p_m^{(j)}(n)-\alpha_{mj}p_j^{(j)}(n), \tag{28}$$

where

$$\alpha_{mj}=\frac{\overline{p_m^{(j)}(n)p_j^{(j)}(n)}}{\overline{(p_j^{(j)}(n))^2}} \tag{29}$$

The $p_j^{(j)}(n)$ in the modified Gram-Schmidt procedure are the orthogonal functions which take the place of the $w_j(n)$ in the original Gram-Schmidt process. Thus, it is easily shown that the m.s.e. in (15) is minimized when

$$g_m=\frac{\overline{y(n)p_m^{(m)}(n)}}{\overline{(p_m^{(m)}(n))^2}}, \quad m=0,\ldots,M \tag{30}$$

and the a_m are calculated from (14).

To obtain the α_{mj} and g_m from (29), (30) efficiently, define for $j=0,\ldots,M$, $m,k=j,\ldots,M$

$$D(m,k,j)=\overline{p_m^{(j)}(n)p_k^{(j)}(n)}, \tag{31}$$

$$C(m,j)=\overline{y(n)p_m^{(j)}(n)}. \tag{32}$$

Note

$$D(m,k,j)=D(k,m,j).$$

Using (28), and an analogous equation obtained by replacing m with k, and also (29), we obtain

$$\overline{p_m^{(j+1)}(n)p_k^{(j+1)}(n)}=\overline{p_m^{(j)}(n)p_k^{(j)}(n)}-\alpha_{kj}\overline{p_m^{(j)}(n)p_j^{(j)}(n)} \tag{33}$$

or equivalently

$$\begin{aligned}&D(m,k,j+1)=D(m,k,j)-\alpha_{kj}D(m,j,j),\\ &j=0,\ldots,M-1, \quad m,k=j+1,\ldots,M.\end{aligned} \tag{34}$$

Moreover, it follows from (3), (27), and (31) that

$$D(0,0,0)=1 \tag{35}$$

and for $m,k=1,\ldots,M$

$$D(m,0,0)=\overline{p_m(n)}, \tag{36}$$

$$D(m,k,0)=\overline{p_m(n)p_k(n)}, \tag{37}$$

and from (29)

$$\begin{aligned}&\alpha_{mj}=\frac{D(m,j,j)}{D(j,j,j)},\\ &j=0,\ldots,M-1, \quad m=j+1,\ldots,M.\end{aligned} \tag{38}$$

Next, from (28),

$$\overline{y(n)p_m^{(j+1)}(n)}=\overline{y(n)p_m^{(j)}(n)}-\alpha_{mj}\overline{y(n)p_j^{(j)}(n)}$$

and hence from (32)

$$\begin{aligned}&C(m,j+1)=C(m,j)-\alpha_{mj}C(j,j),\\ &j=0,\ldots,M-1, \quad m=j+1,\ldots,M.\end{aligned} \tag{39}$$

Moreover, from (3), (27), and (32),

$$C(0,0)=\overline{y(n)}, \tag{40}$$

$$C(m,0)=\overline{y(n)p_m(n)}, \quad m=1,\ldots,M. \tag{41}$$

And, from (30)–(32)

$$g_m=\frac{C(m,m)}{D(m,m,m)}, \quad m=0,\ldots,M. \tag{42}$$

We first use (34)–(38) to find the α_{mj}, for example using the following pseudocode:

$D(0,0,0)=1$

FOR $m=1$ TO M

COMPUTE $D(m,0,0)$ FROM (36)

FOR $k=1$ TO m

Compute $D(m,k,0)$ FROM (37)

NEXT k

NEXT m

FOR $j=0$ TO $M-1$

FOR $m=j+1$ TO M

COMPUTE α_{mj} FROM (38)

FOR $k=j+1$ TO m

COMPUTE $D(m,k,j+1)$ FROM (34)

NEXT k

NEXT m

NEXT j.

In fact, this pseudocode yields both the α_{mj} and the $D(m,m,m)$. This enables us to employ (39)–(42) to find all the g_m. Then (14) can be used to find the a_m in (2).

5.1 System Identification

Recall that the orthogonal functions $p_j^{(0)}(n)$ correspond to the $w_j(n)$ considered in the original Gram-Schmidt process. Therefore, by (6) and (31), the reduction in m.s.e. from adding term $a_M p_M(n)$ to the model is

$$Q(M)=g_M^2 D(M,M,M). \tag{43}$$

Thus, to expand the model by a further term using the robust orthogonal search, evaluate the quantity Q for each candidate term, and select the candidate with greatest Q value (optionally, subject to exceeding a specified threshold level). Continuing in this way, we can rapidly build up an accurate parsimonious model for a system of unknown structure. Clearly many similar strategies utilizing (43) to select terms can be set down: e.g. considering two or more candidates at a time.

Finally, note that, analogous to (16), the m.s.e. after adding $a_M p_M(n)$ to the model is

$$\text{m.s.e.}=\overline{y^2(n)}-\sum_{m=0}^{M} g_m^2 D(m,m,m). \tag{44}$$

The above pseudocode following (42) can be shortened to avoid repetition of calculation already performed at earlier stages of the searching. Thus, suppose that model terms up to $m=M-1$ have been selected. Then in testing each candidate for $p_M(n)$, the above pseudocode merely need be carried out with $m=M$:

$D(0,0,0)=1$

COMPUTE $D(M,0,0)$ FROM (36)

FOR $k=1$ TO M

COMPUTE $D(M,k,0)$ FROM (37)

NEXT k

For $j=0$ TO $M-1$

COMPUTE α_{Mj} FROM (38)

FOR $k=j+1$ TO M

COMPUTE $D(M,k,j+1)$ FROM (34)

NEXT k

NEXT j.

We then set $m=M$ in (39), (41), and (42) to obtain g_M and thus measure the benefit of the candidate term via (43). After the choice for $p_M(n)$ has been made, we carry out the immediately above pseudocode once again for the chosen candidate, and also redetermine g_M. This is done to set the values of α_{Mj}, $D(M,M,M)$, $C(M,M)$ and g_M prior to either initiating the search for $p_{M+1}(n)$ or stopping the process. Note that (44) can be used to evaluate the m.s.e. in deciding when to terminate the model development. In addition, a threshold level may be set as the minimum reduction in mean-square error required before a further addition can be made to the model.

5.2 Time-Series Analysis

We next let $y(n)$, $n=0,\ldots,N$ be a given time-series. Let (2) represent a sinusoidal series model which is to be constructed for the time-series using the robust orthogonal search. The procedure is analogous to the time-series application of the fast orthogonal search set out in Sect. 3.2, and for example, (19)–(21) still hold. Adding the i-th term pair on the right side of (21) to the model reduces the mean-square error by the amount

$$Q_1(i)=g_{2i-1}^2 D(2i-1,2i-1,2i-1)+g_{2i}^2 D(2i,2i,2i). \tag{45}$$

This equation is used in place of (22). Again, at the stage of adding the i-th term pair, $M=2i$.

To evaluate $Q_1(i)$ in (45) for a candidate frequency requires knowing the corresponding g_m, $D(m,m,m)$ when $m=2i-1$ and $m=2i$. For a given candidate frequency, use the pseudocode immediately following (42) with $M=2i$, and obtain α_{mj} and $D(m,m,m)$, $m=M-1$, M; $j=0,\ldots,m-1$. Note that the pseudocode merely need be carried out for $m \geqq M-1$ and not for earlier values of m. This is similarly true in using (39), (41), (42), to obtain $C(m,m)$ and thence g_m, $m=M-1$, M, in order to evaluate $Q_1(i)$ for the candidate. Similarly obtain $Q_1(i)$ for all other candidate frequencies as yet unselected. Finally set ω_i equal to the candidate for which $Q_1(i)$ is greatest (optionally, subject to exceeding a threshold level). Also ensure that $D(m,m,m)$, $m=M-1$, M, is not negligible. Using the chosen frequency for ω_i, redetermine α_{mj}, D(m, m, m), C(m, m) and g_m, $m=M-1$, M; $j=0,\ldots,m-1$.

Next, evaluate the mean-square error via (44). Stop the procedure if the mean-square error is acceptably small, or if a pre-set limit on the number of model terms has been reached. Model development may also be terminated if no remaining candidate frequency can cause a reduction in mean-square error exceeding a specified threshold level. These criteria, optionally in conjunction with an F-test, determine the final value for M (i.e. the model order) in the sinusoidal series represented by (2). Equation (2) may then serve as an economical sinusoidal series approximation revealing the significant frequencies in the original time-series data.

6 Discussion and Conclusion

The orthogonal search method (Korenberg 1985; McIlroy 1986) can be applied in many variations. For example to increase speed, McIlroy (1986) arranged candidate difference equation terms into disjoint subsets of (1) linear x terms, (2) linear y terms, (3) xx terms, (4) yy terms, (5) xy terms. These subsets were searched successively, rather than searching all candidate terms together. Within a given subset, a candidate term was permanently dropped from consideration if its potential reduction of mean-square error was less than a specified percentage of the output mean-square. Similar successive searches are readily implemented using the fast and robust orthogonal algorithms examined above.

Also note that an early selection of model terms is possible for both system identification and time-series analysis. For example in the latter application, when searching for the first sinusoidal frequency, we evaluated $Q_1(1)$ as a function of each candidate frequency. We can use this function to select the model frequencies at once. Simply choose the candidate frequency with largest $Q_1(1)$ value and all candidate frequencies occurring at "relative maxima" of $Q_1(1)$, which exceed a specified threshold level. A similar strategy can be used to choose difference equation terms, e.g. by first arranging them into subsets such as *(1)* linear x terms, in order, *(2)* linear y terms, in order, *(3)* terms of subset (1) multiplied by $x(n)$, *(4)* terms of subset (2) multiplied by $x(n)$, and other remaining subsets. Then evaluate $Q(1)$ over the first subset and select the terms in the subset occurring at the absolute maximum and "relative maxima" which exceed a threshold level. The next subset is searched in an analogous fashion, and so on.

The fast and the robust orthogonal searches can both be used to identify difference equation models of many biological systems, without a priori knowledge of system structure. With or without the searching feature, they enable accurate identification of kernels in functional expansions of nonlinear systems (Korenberg 1988). The robust method of carrying out the Cholesky factorization is, moreover, suitable for accurate kernel estimation of systems with long memory length. This accuracy enables a more precise identification of cascades of alternating dynamic linear and static nonlinear systems.

For example, if first and second linear systems, with transfer functions $G_1(\omega)$ and $G_2(\omega)$ respectively, are separated by a static nonlinearity, then (Korenberg 1973)

$$|G_1(\omega)| = C_1 \frac{|H(\omega, -\omega/2)|}{|H(\omega/2)|}, \tag{46}$$

$$|G_2(\omega)| = C_2 \frac{|H(\omega)|\,|H(\omega/2)|}{|H(\omega, -\omega/2)|}. \tag{47}$$

Here C_1, C_2 are constants and $H(\cdot)$ and $H(\cdot,\cdot)$ are the one- and two-dimensional Fourier transforms of the first- and second-order Wiener kernels respectively. The corresponding phases of the linear systems in the cascade are, under fairly broad conditions,

$$\underline{/G_1(\omega)} = \underline{/H(\omega, \infty)}, \tag{48}$$

$$\underline{/G_2(\omega)} = \underline{/H(\omega)} - \underline{/H(\omega, \infty)}. \tag{49}$$

[In practice, "∞" on the right sides of (48), (49) is replaced by a large positive constant.] Thus, the accurate estimation of Wiener kernels enables the linear systems in the cascade to be identified from (46)–(49). The static nonlinearity is then readily measured either graphically or by least-square fitting (Korenberg 1973; Korenberg et al. 1988b).

The time-series methods (Korenberg 1987, 1988) examined in the present paper provide good resolution of frequencies even with short and noisy data records. Moreover, the required order of the

sinusoidal series model developed is automatically determined, and no polynomial equation must be solved. In future papers, we intend to apply this time-series analysis to EEG, ECG, EMG and other biological time-series data.

Acknowledgements. I particularly wish to thank Dr. Ken-Ichi Naka and Dr. Hiroko M. Sakai of the National Institute for Basic Biology, Okazaki, Japan for their kind support and encouragement. This research was sponsored by the Natural Sciences and Engineering Research Council of Canada, and the Advisory Research Committee of Queen's University.

References

Billings, SA, Leontaritis IJ (1982) Parameter estimation techniques for nonlinear systems. IFAC Symp Ident Sys Param Est 1:427–432

Box GEP, Jenkins GM (1976) Time series analysis: forecasting and control. Holden-Day, San Francisco

Cooley JW, Tukey JW (1965) An algorithm for machine calculation of complex Fourier series. Math Comput 19:297–301

Dwight HB (1960) Tables of integrals and other mathematical data. Macmillan, New York

French AS, Butz EG (1973) Measuring the Wiener kernels of a nonlinear system using the fast Fourier transform algorithm. Int J Control 17:529–539

Haber R, Keviczky L (1976) Identification of nonlinear dynamic systems. IFAC Symp Ident Sys Param Est 1:79–126

Ho T, Kwok J, Law J, Leung L (1987) Nonlinear system identification. Project Rept ELEC-490, Department of Electrical Engineering, Queen's University, Kingston, Ontario, Canada

Kay SM, Marple SL (1981) Spectrum analysis – a modern perspective. Proc. IEEE 69:1380–1419

Korenberg MJ (1973) Identification of biological cascades of linear and static nonlinear systems. Proc Midwest Symp Circuit Theory 18.2:1–9

Korenberg MJ (1985) Orthogonal identification of nonlinear difference equation models. Proc Midwest Symp Circuit Sys 1:90–95

Korenberg MJ (1987) Fast orthogonal identification of nonlinear difference equation and functional expansion models. Proc Midwest Symp Circuit Sys 1:270–276

Korenberg MJ (1988) Identifying nonlinear difference equation and functional expansion representations: the fast orthogonal algorithm. Ann Biomed Eng 16:123–142

Korenberg MJ, Bruder SB, McIlroy PJ (1988a) Exact orthogonal kernel estimation from finite data records: extending Wiener's identification of nonlinear systems. Ann Biomed Eng 16:201–214

Korenberg MJ, French AS, Voo SKL (1988b) White-noise analysis of nonlinear behavior in an insect sensory neuron: kernel and cascade approaches. Biol Cybern 58:313–320

Lee YW, Schetzen M (1965) Measurement of the Wiener kernels of a nonlinear system by cross-correlation. Int J Control 2:237–254

Marmarelis PZ, Marmarelis VZ (1978) Analysis of physiological systems. The white noise approach. Plenum Press, New York

Marmarelis PZ, Naka K-I (1972) White noise analysis of a neuron chain: an application of the Wiener theory. Science 175:1276–1278

McIlroy PJH (1986) Applications of nonlinear systems identification. MSc Thesis, Queen's University, Kingston, Ontario, Canada

Mohanty NC (1986) Random signals estimation and identification. Analysis and applications. Van Nostrand, New York

Nugent ST, Finley JP (1983) Spectral analysis of periodic and normal breathing in infants. IEEE Trans Biomed Eng 30:672–675

Palm G, Poggio T (1978) Stochastic identification methods for nonlinear systems: an extension of the Wiener theory. SIAM J Appl Math 34:524–534

Rice JR (1966) A theory of condition. SIAM J Numer Anal 3:287–310

Sterman MB (1981) Power spectral analysis of EEG characteristics during sleep in epileptics. Epilepsia 22:95–106

Wiener N (1958) Nonlinear problems in random theory. Wiley, New York

Received: June 27, 1988

Michael J. Korenberg
Department of Electrical Engineering
Queen's University
Kingston, Ontario
K7L 3N6 Canada