A Review of Parametric Modelling Techniques for EEG Analysis

J. Pardey, S. Roberts*, L. Tarassenko

University of Oxford, Medical Engineering Unit, 43 Banbury Road, Oxford OX2 6PE

***Imperial College of Science, Technology & Medicine, Exhibition Road, London SW7 2BT**

ABSTRACT — *This tutorial provides an introduction to the use of parametric modelling techniques for time series analysis, and in particular the application of autoregressive modelling to the analysis of physiological signals such as the human electroencephalogram. The concept of signal stationarity is considered and, in the light of this, both adaptive models, and non–adaptive models employing fixed or adaptive segmentation, are discussed. For non–adaptive autoregressive models, the Yule–Walker equations are derived and the popular Levinson–Durbin and Burg algorithms are introduced. The interpretation of an autoregressive model as a recursive digital filter and its use in spectral estimation are considered, and the important issues of model stability and model complexity are discussed.*

Keywords: autoregressive modelling, biomedical signal processing, human sleep EEG

1 INTRODUCTION

Parametric modelling is a technique for time series analysis in which a mathematical model is fitted to a sampled signal. If the model forms a good approximation to the signal's observed behaviour it can then be used in a wide range of applications, such as spectral estimation, linear prediction coding (LPC) for data compression, speech synthesis, and feature extraction for pattern classification problems.

The mathematical model that is most widely used is a rational transfer function, the exact form of which is determined by estimating suitable values for its free parameters. If all of these parameters lie in the transfer function's denominator then the model is termed an all–pole or *autoregressive* (AR) model, while an all–zero or *moving–average* (MA) model has all of its free parameters in the numerator. A model with free parameters in both the numerator and denominator is then termed a pole–zero or *autoregressive moving–average* (ARMA) model.

Furthermore, in *adaptive* models the values of the free parameters are updated with the arrival of each

new data sample, whereas in *non–adaptive* models the parameters are chosen so as to give the best fit to a sequence of data samples. Because of this, non–adaptive models require that the signal is *stationary*, i.e. that its statistical characteristics, such as average amplitude and frequency content, do not vary with time. Most signals, including speech and the electroencephalogram (EEG), are non–stationary (i.e. they have a time–varying frequency spectrum), although they can be considered locally stationary over short time intervals. For such signals, either an adaptive model can be used, or the signal can be divided into sufficiently short, quasi–stationary segments and a non–adaptive model fitted to each segment. The length of these segments can be either fixed, typically at one second for EEG analysis, or variable, in which case the signal is continuously monitored for departures from stationarity and segment boundaries are placed accordingly.[1]

The key to the performance of parametric modelling techniques, however, lies in the relative effectiveness of the various algorithms that can be used to estimate the free parameters. For non–adaptive AR models the two most popular algorithms are the Levinson–Durbin algorithm and the Burg algorithm, while for adaptive AR models the Kalman filtering algorithm[2,3] is commonly used. This is summarised in Figure 1.

The relative simplicity and reliability of the Levinson–Durbin and Burg algorithms has made non–adaptive AR modelling by far the most popular method of time series analysis to date, and it is this method that will be considered in the remainder of the paper. The development of these algorithms in such diverse areas as economic forecasting and geophysics, however, has led to confusion both in the terminology used and in the different perspectives from which the algorithms are derived. One purpose of this paper is thus to streamline the approach to AR modelling and algorithm development. The AR modelling technique is formulated in Section 2, where the Yule–Walker equations are derived and the Levinson–Durbin and Burg algorithms are presented. In Section 3 the interpretation of the AR model as a recursive digital filter, its use in spectral estimation, and its stability are considered, while the choice of model complexity is investigated in Section 4. Section 5 summarises the material presented in Sections 2 to 4, and concludes with a few comments on adaptive AR modelling.

2 AUTOREGRESSIVE MODELLING

The AR modelling technique can be formulated either in the frequency domain as a spectral matching problem or in the time domain as a linear prediction problem.[4] The latter approach, which is more intuitive and will therefore be adopted here, assumes that the value of the current sample, s_n, in a data sequence, $s_1, s_2, \ldots, s_N$, can be predicted as a linearly weighted sum of the p most recent sample values, $s_{n-1}, s_{n-2}, \ldots, s_{n-p}$, where p is the model order and is generally chosen to be much smaller than the sequence length, N. If $\tilde{s}_n$ denotes the predicted value of s_n, then this can be expressed as follows:

$$\tilde{s}_n \;=\; -\sum_{i=1}^{p} a_{pi} s_{n-i} \tag{1}$$

where the weight, a_{pi}, denotes the ith coefficient of the pth–order model. This is depicted in Figure 2. The error between the actual value and the predicted value is called the forward prediction error, e_{pn}, and is given by:

$$e_{pn} \;=\; s_n - \tilde{s}_n \;=\; s_n + \sum_{i=1}^{p} a_{pi} s_{n-i} \tag{2}$$

The mean of the squared prediction errors for the entire data sequence, $s_1, s_2, \ldots, s_N$, is equal to the prediction error power, E (assuming the missing samples prior to s_1 to be zero in the calculation of $\tilde{s}_1, \tilde{s}_2, \ldots, \tilde{s}_p$):

$$E \;=\; \frac{1}{N}\sum_{n=1}^{N} e_{pn}^2 \;=\; \frac{1}{N}\sum_{n=1}^{N} (s_n - \tilde{s}_n)^2 \;=\; \frac{1}{N}\sum_{n=1}^{N} (s_n + \sum_{i=1}^{p} a_{pi} s_{n-i})^2 \tag{3}$$

Note that the use of (1) assumes that the signal being modelled is linear, even though the process (or processes) generating it may be non–linear, in which case the use of non–linear methods to model the signal would perhaps be more appropriate. For applications in EEG analysis, however, a comparison of non–linear forecasting methods versus the predictive performance — as given by (2) — of AR modelling techniques has shown that the latter gives very similar, or even slightly improved performance over non–linear methods.[5]

Given that the technique is a suitable one, therefore, an estimate is required of the coefficients, $a_{p1}, a_{p2}, \ldots, a_{pp}$. Typically a least–squares error criterion is used, whereby the best fit of the pth–order model in (1) to a given data sequence is obtained by finding the set of coefficients for which E in (3) is minimised. This is achieved by setting:

$$\frac{\partial E}{\partial a_{pi}} \;=\; 0, \qquad \text{for } 1 \le i \le p$$

which yields the following set of p equations in p unknowns:

$$\begin{aligned}
\left(\frac{1}{N}\sum_{n=1}^{N} s_{n-1}s_{n-1}\right) a_{p1} \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-2}s_{n-1}\right) a_{p2} \;+\; \cdots \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-p}s_{n-1}\right) a_{pp} \;&=\; -\left(\frac{1}{N}\sum_{n=1}^{N} s_n s_{n-1}\right) \\
\left(\frac{1}{N}\sum_{n=1}^{N} s_{n-1}s_{n-2}\right) a_{p1} \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-2}s_{n-2}\right) a_{p2} \;+\; \cdots \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-p}s_{n-2}\right) a_{pp} \;&=\; -\left(\frac{1}{N}\sum_{n=1}^{N} s_n s_{n-2}\right) \\
\vdots \qquad\qquad\qquad\qquad \vdots \qquad\quad \vdots \qquad\qquad\qquad\qquad \vdots \qquad\qquad\qquad\quad & \\
\left(\frac{1}{N}\sum_{n=1}^{N} s_{n-1}s_{n-p}\right) a_{p1} \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-2}s_{n-p}\right) a_{p2} \;+\; \cdots \;+\; \left(\frac{1}{N}\sum_{n=1}^{N} s_{n-p}s_{n-p}\right) a_{pp} \;&=\; -\left(\frac{1}{N}\sum_{n=1}^{N} s_n s_{n-p}\right)
\end{aligned} \tag{4}$$

Solving these for $a_{p1}, a_{p2}, \ldots, a_{pp}$ and substituting the values obtained back into (3) gives an expression for the minimum prediction error power, denoted by E_p:

$$E_p \;=\; \left(\frac{1}{N}\sum_{n=1}^{N} s_n^2\right) \;+\; \sum_{i=1}^{p}\left(\frac{1}{N}\sum_{n=1}^{N} s_n s_{n-i}\right) a_{pi} \tag{5}$$

However, given that the autocorrelation function of an infinite data sequence, $s_{-\infty}, \ldots, s_{\infty}$, is given by:

$$R_i \;=\; \lim_{N\to\infty} \frac{1}{N}\sum_{n=1}^{N} s_n s_{n-i}, \qquad \text{for } -\infty < i < \infty$$

where $R_i = R_{-i}$ (i.e. an even function of i) if the data sequence is stationary, so the parenthesised terms in (4) and (5) are just estimates of the first $p+1$ terms of the truncated autocorrelation function, $R_0, R_1, \ldots, R_p$, using only the finite data sequence, $s_1, s_2, ..., s_N$:

$$\tilde{R}_{|i-j|} \;=\; \frac{1}{N}\sum_{n=1}^{N} s_{n-i}s_{n-j}, \qquad \text{for } 0 \le i \le p \text{ and } 1 \le j \le p \tag{6}$$

where the missing samples prior to s_1 are assumed to be zero, as in (3). The use of $|i-j|$ indicates that $\tilde{R}_{|i-j|}$ depends only on the difference between i and j (assuming stationarity) and not on their individual values. Substituting the values of $\tilde{R}_0, \tilde{R}_1, \ldots, \tilde{R}_p$ obtained using (6) into (4) and (5), and rearranging (4) in matrix form gives:

$$\begin{bmatrix} \tilde{R}_0 & \tilde{R}_1 & \cdots & \tilde{R}_{p-1} \\ \tilde{R}_1 & \tilde{R}_0 & \cdots & \tilde{R}_{p-2} \\ \vdots & \vdots & & \vdots \\ \tilde{R}_{p-1} & \tilde{R}_{p-2} & \cdots & \tilde{R}_0 \end{bmatrix} \begin{bmatrix} a_{p1} \\ a_{p2} \\ \vdots \\ a_{pp} \end{bmatrix} \;=\; -\begin{bmatrix} \tilde{R}_1 \\ \tilde{R}_2 \\ \vdots \\ \tilde{R}_p \end{bmatrix}$$

$$E_p \;=\; \tilde{R}_0 \;+\; \sum_{i=1}^{p} a_{pi}\tilde{R}_i \tag{7}$$

Alternatively, the matrix equation in (7) can be augmented to include the expression for E_p:

$$\begin{bmatrix} \tilde{R}_0 & \tilde{R}_1 & \tilde{R}_2 & \cdots & \tilde{R}_p \\ \tilde{R}_1 & \tilde{R}_0 & \tilde{R}_1 & \cdots & \tilde{R}_{p-1} \\ \tilde{R}_2 & \tilde{R}_1 & \tilde{R}_0 & \cdots & \tilde{R}_{p-2} \\ \vdots & \vdots & \vdots & & \vdots \\ \tilde{R}_p & \tilde{R}_{p-1} & \tilde{R}_{p-2} & \cdots & \tilde{R}_0 \end{bmatrix} \begin{bmatrix} 1 \\ a_{p1} \\ a_{p2} \\ \vdots \\ a_{pp} \end{bmatrix} = \begin{bmatrix} E_p \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{8}$$

Equations (7) or (8) are called the *Yule–Walker equations*, and describe the p unknown AR coefficients in terms of the $p+1$ estimated autocorrelation coefficients. Solving the Yule–Walker equations for $a_{p1}, a_{p2}, ..., a_{pp}$ is termed the *autocorrelation method* of AR parameter estimation and can be accomplished using either a standard technique such as Gaussian elimination,[6] or a recursive technique such as the *Levinson–Durbin algorithm*.[4] The latter approach is computationally more efficient since it exploits the fact that the autocorrelation matrix on the left–hand side of (7) or (8) is both symmetric and a Toeplitz matrix (i.e. the terms along any diagonal are the same). The algorithm, which is shown in Figure 3, solves the Yule–Walker equations for each value of the model order from $m = 0$ to $m = p$. On each pass through the algorithm the estimated autocorrelation coefficients are used to generate a single, new coefficient, a_{mm}. The remaining coefficients, $a_{m1}, a_{m2}, \ldots, a_{m(m-1)}$, are then generated recursively from their $(m-1)$th–order values, $a_{(m-1)1}, a_{(m-1)2}, \ldots, a_{(m-1)(m-1)}$, which are known from the previous pass through the algorithm. The expression for a_{mi} in Figure 3 is called the *Levinson recursion*, and will be used again later on.

The algorithm thus calculates the parameter sets, $\{E_0\}, \{a_{11}, E_1\}, \{a_{21}, a_{22}, E_2\}$, and so on, for all of the lower order fits, $m < p$, to the data until the desired solution, $\{a_{p1}, a_{p2}, \ldots, a_{pp}, E_p\}$, is obtained. Note that $m = 0$ describes a zeroth–order model which does no prediction at all, so that E_0 is simply the power in the data sequence, $s_1, s_2, \ldots, s_N$, and this in turn is equal to the zeroth autocorrelation coefficient, $\tilde{R}_0$, in (6). The intermediate values, k_m, in Figure 3 are called the *reflection*, or *partial correlation* (PARCOR) coefficients, and can be interpreted as the partial correlation between s_n and s_{n+m} holding $s_{n+1}, s_{n+2}, \ldots, s_{n+m-1}$ constant.

Once the coefficients, $a_{p1}, a_{p2}, ..., a_{pp}$, have been obtained, the AR model can be applied to the same data sequence, $s_1, s_2, \ldots, s_N$, but in the reverse direction. This is shown in Figure 4, where the value of the sample, s_{n-p}, is retrospectively "predicted" as a linearly weighted sum of the p *future* samples,

$s_{n-p+1}, s_{n-p+2}, \ldots, s_n$:

$$\tilde{s}_{n-p} \;=\; -\sum_{i=1}^{p} a_{pi} s_{n-p+i}$$

The error between the actual value and the predicted value in this case is called the *backward prediction error*, b_{pn}:

$$b_{pn} \;=\; s_{n-p} - \tilde{s}_{n-p} \;=\; s_{n-p} + \sum_{i=1}^{p} a_{pi} s_{n-p+i} \tag{9}$$

(Note that although this describes the prediction error for s_{n-p} it is denoted by b_{pn} and not $b_{p(n-p)}$ as might intuitively be expected; this peculiar notation is just a mathematical convenience to simplify the expressions that follow.) Furthermore, the fact that the AR coefficients were generated using the Levinson recursion in Figure 3 enables the following recursive relationships to be derived (see Appendix) between the forward prediction error in (2) and the backward prediction error in (9):

$$e_{pn} \;=\; e_{(p-1)n} \;+\; a_{pp} b_{(p-1)(n-1)} \tag{10}$$

$$b_{pn} \;=\; b_{(p-1)(n-1)} \;+\; a_{pp} e_{(p-1)n} \tag{11}$$

These relationships express the pth–order prediction errors for s_n and s_{n-p} in terms of their corresponding $(p-1)$th–order prediction errors, and lead to a second, superior technique for AR parameter estimation called the *maximum entropy method* (MEM). Like the autocorrelation method described above, the maximum entropy method is a recursive estimation technique based on a least–squares error criterion. However, in the derivation of the Yule–Walker equations, the range of the summation in the expressions for E in (3) and $\tilde{R}_{|i-j|}$ in (6) implicitly assumes that the data outside the interval, $s_1, s_2, \ldots, s_N$, are zero. Since this is almost always an unrealistic assumption, the maximum entropy method restricts the range of the summation so as to use only the available data. Furthermore, instead of minimising only the forward prediction error power, the maximum entropy method seeks to minimise the mean of both the forward and backward prediction error powers:

$$E \;=\; \frac{1}{2(N-p)} \sum_{n=p+1}^{N} (e_{pn}^2 + b_{pn}^2) \tag{12}$$

subject to the constraint that the AR coefficients are updated using the Levinson recursion. This constraint enables the recursive relationships in (10) and (11) to be used, so that (12) can be expanded as follows:

$$E \;=\; \frac{1}{2(N-p)} \sum_{n=p+1}^{N} \left(\left[e_{(p-1)n} + a_{pp} b_{(p-1)(n-1)} \right]^2 + \left[b_{(p-1)(n-1)} + a_{pp} e_{(p-1)n} \right]^2 \right)$$

This is a function of the unknown coefficient, a_{pp}, and the $(p-1)$th–order forward and backward prediction errors, which are known from the previous pass through the algorithm. E can thus be minimised by setting:

$$\frac{dE}{da_{pp}} \;=\; 0$$

which yields

$$a_{pp} \;=\; -2 \sum_{n=p+1}^{N} b_{(p-1)(n-1)} e_{(p-1)n} \;\Bigg/\; \sum_{n=p+1}^{N} [b^2_{(p-1)(n-1)} + e^2_{(p-1)n}] \qquad (13)$$

Using (13) in place of the corresponding expression in the Levinson–Durbin algorithm and adding the extra recursions for e_{pn} and b_{pn} yields the *Burg algorithm*[7] shown in Figure 5. An additional step can also be included in the Burg algorithm that reduces the computational complexity of (13) by calculating the denominator recursively:[8]

$$\mathrm{den}_p \;=\; \mathrm{den}_{p-1}[1 - a^2_{(p-1)(p-1)}] \;-\; b^2_{(p-1)(N-p)} \;-\; e^2_{(p-1)p}$$

where $\mathrm{den}_0 = 2E_0 N$ from (12) and (13).

3 SPECTRAL ESTIMATION AND MODEL STABILITY

By rearranging the expression for the forward prediction error in (2) the AR model can be viewed as an all–pole, or infinite–impulse–response (IIR) filter whose current output, s_n, is a function of both the p most recent outputs, $s_{n-1}, s_{n-2} \ldots, s_{n-p}$, and the current input, e_{pn}:

$$s_n \;=\; \tilde{s}_n + e_{pn} \;=\; -\sum_{i=1}^{p} a_{pi} s_{n-i} + e_{pn} \qquad (14)$$

This is shown in Figure 6. For applications such as EEG analysis, where the output signal is the observed EEG, the input signal is inaccessible and hence unknown. However, if the assumption made at the beginning of the previous section is correct (i.e. that s_n is predictable from a linearly weighted sum of $s_{n-1}, s_{n-2}, \ldots, s_{n-p}$), then the predicted values, $\tilde{s}_1, \tilde{s}_2, \ldots, \tilde{s}_N$, can be interpreted as the true, underlying signal, while the actual values, $s_1, s_2, \ldots, s_N$, can be regarded as these predicted values corrupted by additive white noise which, being uncorrelated and therefore unpredictable, gives rise to the prediction errors, $e_{p1}, e_{p2}, \ldots, e_{pN}$. The assumption just referred to can thus be re–phrased with respect to Figure 6, in which it is assumed that the output sequence, $s_1, s_2, \ldots, s_N$, is the result of using a pth–order AR model to filter a white noise input sequence, $e_{p1}, e_{p2}, \ldots, e_{pN}$. It follows that when fitting an AR model

to a data sequence, any departure of the prediction errors away from a white noise sequence can be used to indicate the goodness of fit of the model to the signal.

Despite this observation, it is a commonly held misconception that the application of AR modelling to EEG analysis is useful even if the prediction errors are correlated, since they can then be interpreted as the underlying "input signal" which, when filtered by the AR model, produces the observed EEG. This physiological interpretation of the AR model — whereby both the filter characteristics and the input signal are simultaneously revealed — is clearly incorrect, however, since a correlated sequence of prediction errors simply reflects the poor fit of the model to the data.

Spectral estimation

The AR filter described by (14) can be specified in the frequency domain by taking the z–transform of the original expression in (2). If $E(z)$ and $S(z)$ are the z–transforms of $e_{p1}, e_{p2}, \ldots, e_{pN}$ and $s_1, s_2, \ldots, s_N$ respectively, then:

$$E(z) = A(z)S(z), \quad \text{where } A(z) = 1 + \sum_{i=1}^{p} a_{pi} z^{-i}$$

$$A^{-1}(z) = \frac{S(z)}{E(z)} = 1 \Big/ (1 + \sum_{i=1}^{p} a_{pi} z^{-i}) \tag{15}$$

$A^{-1}(z)$ is the AR model's transfer function, usually denoted by $H(z)$. Its frequency response, $H(\omega)$, is determined by evaluating $H(z)$ along the unit circle in the z–plane, where $z = \mathrm{e}^{j\omega T}$ for a sampling period, T. Furthermore, if $E(z)$ is a white noise input sequence then its spectrum, $E(\omega)$, will be flat and the spectrum of the output sequence, $S(\omega) = H(\omega)E(\omega)$, will be equal to $H(\omega)$ scaled by the constant, $E(\omega) = E_p T$. In practice, however, $E(z)$ only approximates a white noise sequence and so $S(\omega)$ can only be estimated. This estimate, $\tilde{S}(\omega)$, is given by:

$$\tilde{S}(\omega) = E_p T \Big/ \left| 1 + \sum_{i=1}^{p} a_{pi} e^{-ij\omega T} \right|^2 \tag{16}$$

The assumption on which the AR modelling technique is based can now be re–phrased in the frequency domain, where it is assumed that the flat spectrum of the white noise input sequence is "coloured" by the AR model to produce an output spectrum of the desired shape. Factorisation of the denominator in (16) also reveals that depending on the values of the AR coefficients, the denominator may be zero (corresponding to infinite power) at certain, discrete frequencies. This makes AR modelling particularly

suited to the types of signal that occur in nature, such as speech, EEG, and seismic data, since these tend to be characterised by their dominant frequencies (i.e. sharply defined spectral peaks), rather than by the absence of power at certain frequencies (spectral notches) which can be shown to be better approximated by an MA model.[9] The more general ARMA model is appropriate if the spectrum is thought to contain both peaks and notches, although this requires an additional set of coefficients to be estimated for the MA part and involves the solution of complicated non–linear equations.[9,10]

AR spectral estimation often gives a very significant improvement in frequency resolution compared to the traditional periodogram method as implemented by the fast Fourier transform (FFT).[6] The estimated AR spectrum of a data sequence, $s_1, s_2, \ldots, s_N$, is a continuous function of frequency and can thus be evaluated numerically at any number of frequencies — uniformly spaced or otherwise — in the interval, $0.0 \leq f \leq 0.5$ (where f is normalised with respect to the sampling frequency). Conversely the periodogram is a discrete spectrum, evaluated only at the N uniformly spaced (i.e. harmonically related) frequencies, $f_n = n/N$, where $n = 0, 1, \ldots, N-1$. The spacing between these frequencies is therefore determined by the sequence length, N, and if this spacing is large (i.e. the sequence length is short) the periodogram may fail to resolve spectral peaks that are close together. Application of the periodogram method to non–stationary signals such as the EEG thus involves a trade–off between the requirements of a short sequence length to ensure stationarity and a long sequence length to ensure good frequency resolution. The FFT additionally requires N to be a power of two (unlike the Levinson–Durbin and Burg algorithms), although this constraint is less of a problem in practice.

For short sequence lengths, the sparseness of the frequencies, f_n, in the periodogram also makes the shape of the spectrum difficult to discern, particularly if these frequencies do not coincide with the dominant frequencies in the signal. Such ambiguity can be avoided by augmenting the N original data samples with extra zeroes. The number of zeroes must be such that the extended sequence length is still a power of two, as required by the FFT, so typically N, $3N$, or $7N$ zeroes are used. Zero padding smoothes the spectrum by interpolating extra frequency values between the N unpadded values, although it *does not* improve the underlying frequency resolution.[10]

To illustrate these points, Figure 7(a) shows the ideal spectrum of a data sequence consisting of three superimposed sinusoids at frequencies of 0.10, 0.20, and 0.21 times the sampling frequency, corrupted by wide–band coloured noise. Figures 7(b)–(f) are estimates of this spectrum obtained from 64 samples of

the data sequence, the values of which are tabulated elsewhere.[10]

The periodogram in Figure 7(b) was generated using a 256–point FFT in which the original 64 data samples were appended with 192 zeroes. The periodogram has failed to resolve the two sinusoids at 0.20 and 0.21, and is heavily distorted by sidelobe leakage. The latter effect is due to the inherent rectangular windowing of the data sequence by the FFT, which makes the unrealistic assumption that samples outside the sequence are zero. Indeed, the main lobe of the smaller spectral peak at 0.10 in Figure 7(b) is almost obscured by sidelobe leakage from the larger peak at 0.20. Sidelobe leakage can be reduced by applying a symmetric, tapered window — such as a Hamming or Hanning window[11] — to the data sequence prior to performing the FFT, although this unfortunately reduces the frequency resolution of the periodogram still further. Figure 7(c) shows the 256–point FFT obtained when a Hamming window is applied to the original 64 samples prior to zero padding.

The spectra in Figures 7(d)–(f) were obtained by fitting a 14th–order AR model to the data sequence. In Figure 7(d) the Levinson–Durbin algorithm was used to estimate the AR parameters, but since the autocorrelation method makes the same zero–valued assumption for data samples outside the sequence as the periodogram method, the spectrum is smeared and the sinusoids at 0.20 and 0.21 are not resolved. The absence of sidelobe leakage from the spectrum is in contrast to the periodogram method, however, and this can be shown to be due to the implied, non–zero extrapolation of the estimated autocorrelation function beyond the values of $\tilde{R}_0, \tilde{R}_1, \ldots, \tilde{R}_p$ used in the Yule–Walker equations.[10] Applying a Hamming window to the data sequence prior to using the Levinson–Durbin algorithm enables all three sinusoids to be resolved, as shown in Figure 7(e). In Figure 7(f) the Burg algorithm was used to estimate the AR parameters. This not only yields the best spectral estimate but also removes the need for windowing, since the maximum entropy method makes no assumptions about samples outside the data sequence.

A less obvious advantage of AR spectral estimation over the periodogram method is that very few cycles or even fractions of a cycle — with a wavelength longer than the sequence length — can often be reliably detected.[10] Also the inclusion of a noise term, e_{pn}, in the AR model means that the estimated spectrum is smooth, since its shape depends only on the values of $a_{p1}, a_{p2}, \ldots, a_{pp}$ used to model the signal. The absence of a noise term in the periodogram method means that both the signal and noise are fitted, so that to smooth out random fluctuations in the raw periodograms due to noise, some form of averaging (e.g. over consecutive, usually overlapping segments) must be used. The main advantage of

the FFT over AR spectral estimation is its computational efficiency.

Stability

The interpretation of an AR model as an IIR filter raises the question of its potential instability. This depends on the values of $a_{p1}, a_{p2}, \ldots, a_{pp}$ generated by the Levinson–Durbin or Burg algorithm, and although both algorithms are guaranteed to yield algebraically stable models, numerical instability can still arise due to the accumulation of round–off errors in finite word length computations. The condition for the stability of an AR model is the same as for an IIR filter, namely that the poles of $H(z)$ in (15) — which correspond to the roots of the polynomial, $A(z)$, in its denominator — all lie on or inside the z–plane's unit circle. Whether this condition is met can be established using a standard numerical technique, such as Laguerre's method,[6] to solve the transfer function's characteristic equation:

$$1 + \sum_{i=1}^{p} a_{pi} z^{-i} = 0 \tag{17}$$

but this is computationally expensive. It can be shown, however, that an alternative condition for an AR model's stability is that the magnitude of each reflection coefficient is less than or equal to unity:[4]

$$|k_m| \leq 1, \quad \text{for } 1 \leq m \leq p$$

Inspection of the algorithms in Figures 3 and 5 reveals that an equivalent condition for stability is that the prediction error power is non–negative:

$$E_m \geq 0, \quad \text{for } 1 \leq m \leq p$$

The stability of the model can thus be monitored during the execution of the Levinson–Durbin or Burg algorithm at no extra cost. An unstable model can then be made stable by finding the roots of (17) and either moving the unstable roots onto the unit circle or reflecting them across it, before finally reconstructing the modified AR coefficients. An unstable root, z_i, is moved onto the unit circle using $z_i \rightarrow z_i / |z_i|$, or reflected across the unit circle using $z_i \rightarrow 1/z_i^*$, where z_i^* is the complex conjugate of z_i. The latter solution has the advantage that the magnitude of the frequency response, as given by (16), remains the same.

4 MODEL ORDER ESTIMATION

An issue that is of central importance to the successful application of AR modelling is the selection of an appropriate value for the model order, p. This depends upon both the subsequent application and the complexity of the signal from one segment to the next. In spectral estimation, for example, the accuracy of the estimated spectrum is critically dependent upon the model order that is chosen. Enough poles must be used to resolve all of the peaks in the spectrum (two poles per sinusoid) with additional poles added to provide general spectral shaping and to approximate any notches in the spectrum. Too high a value of model order over–fits the signal and introduces spurious detail such as false peaks into the spectrum, whereas too low a value produces a spectrum that is over–smoothed. Alternatively, the model order required for dimensionality reduction in pattern classification problems depends upon such factors as the distance in input space between the pth–dimensional patterns for each class and their degree of overlap.

Although the correct model order for a given data sequence is not known in advance, it is desirable to minimise the model's computational complexity by choosing the minimum value of p that adequately represents the signal being modelled. Determining this value is often based upon a goodness–of–fit term such as the prediction error power, E_p. In this respect, the recursive nature of the Levinson–Durbin and Burg algorithms is a particularly useful property, as either algorithm can be used to generate progressively higher order models until the curve defined by $E_1, E_2, \ldots, E_p$ either flattens out or reduces to an acceptable value. Since the fit of the model improves as the model order increases, however, the curve of prediction error power is a non–increasing function of p and the optimum model order is rarely apparent from inspection of the error values alone. For this reason more objective methods for model order estimation have been proposed that combine a goodness–of–fit term with a cost function that penalises some measure of the model's complexity, i.e. some function of p. Such methods include criteria based on predictive performance such as the Akaike information criterion (AIC),[12] the criterion autoregressive transfer function (CAT)[13] and the final prediction error (FPE) criterion.[14] The latter, for example, is defined as:

$$\mathrm{FPE}(p) \quad = \quad \left(\frac{N+p+1}{N-p-1}\right) E_p$$

where the cost function in parenthesis is a monotonically increasing function of p that penalises higher order (i.e. more complex) models. The optimum model order is then the value of p for which FPE(p) is

minimised.

Criteria based on stochastic complexity such as the minimum description length (MDL) criterion[15] and the predictive least–squares (PLS) criterion,[16] have also been proposed, along with others based on singular value decomposition (SVD)[9,17] and Bayesian inference.[18] A good review of these criteria is given elsewhere.[19]

The use of the FPE criterion is illustrated in Figure 8 for an automated sleep analysis system[20,21] in which the coefficients of an AR model form the input features to a neural network. Sections of EEG which were unanimously classified by three human experts as either wakefulness, rapid–eye–movement (REM) sleep, or deep sleep were divided into one–second segments and an AR model was fitted to each segment. In total, 4,800 seconds of each class were collected and the corresponding sets of AR coefficients were used to train and test the neural network.

To determine a suitable value for the AR model order, the Levinson–Durbin algorithm was used to calculate values of $E_1, E_2, \ldots, E_{30}$ for each of the 4,800 seconds from each class. The mean values are plotted for each class as the three solid lines in Figure 8, and it is noticeable that all three curves start to flatten out after about $p = 5$.

The corresponding values of the FPE are plotted as circles in Figure 8 and give optimum model orders, as indicated by the arrows, of 6 for wakefulness, 5 for REM, and 3 for deep sleep. These values should not be considered definitive, however, since calculating the FPE on a second–by–second basis gives the distributions of optimum model order shown in the histograms of Figure 9. These indicate the number of seconds out of 4,800 for which $p = 1, 2, \ldots, 30$ is the optimum model order, and demonstrate that if the values suggested by Figure 8 are used then much of the wakefulness and REM data will be either over–fitted or under–fitted (this is not true of deep sleep, however, since its histogram is sharply peaked). It may thus be more appropriate to use those values that either optimally fit the most data — corresponding to the modes of the distributions in Figure 9 — or over–fit as much data as they under–fit, corresponding to the medians of the distributions.

Figure 10 shows a human–scored hypnogram for a whole night's sleep recording. This divides the recording into thirty–second epochs and then assigns each epoch to one of seven classes using a set of standardised sleep–scoring rules.[22] These classes correspond to wakefulness, movement (when the EEG is too corrupted to be reliably scored), REM, and four stages of progressively deeper sleep. The variation

in optimum model order on a second–by–second basis is plotted below the hypnogram, and shows both the drop in model order associated with the initial descent from wakefulness into deep sleep, and the subsequent rise and fall in model order in phase with the regular waxing and waning of REM and deep sleep.

The above example demonstrates the non–trivial nature of the model order selection problem. Moreover since the number of inputs to a neural network must remain fixed, the model order used for feature extraction must also be fixed, regardless of the non–stationarity of the EEG and of the associated variations in optimum model order with time. A compromise can be found, however, by choosing the value of p that minimises a criterion appropriate for quantifying the neural network's performance: for example, the classification error rate on a cross–validation data set. In practice the use of model order estimation criteria is mainly relevant to applications in spectral estimation, where the optimum model order can be estimated and used on a segment–by–segment basis.

5 DISCUSSION

The purpose of this tutorial has been to provide an intuitive and usable introduction to the very popular technique of autoregressive modelling, and to locate this within the wider framework of parametric modelling techniques in general. The difference between adaptive and non–adaptive modelling was explained, along with the related issues of signal stationarity and signal segmentation. The two most popular and well–established methods for AR parameter estimation are the autocorrelation method, in which the Yule–Walker equations are solved using the Levinson–Durbin algorithm, and the maximum entropy method, as implemented by the Burg algorithm. The correspondence between AR modelling and IIR filtering highlights the need to monitor the model's stability, and also leads to an understanding of its use in spectral estimation. Indeed, the advantages of AR spectral estimation over the FFT are manifold, particularly when using the Burg algorithm and when analysing short data sequences of the kind demanded by non–stationary signals.

A variation on the Burg algorithm can be obtained by removing the constraint that the AR coefficients are updated using the Levinson recursion and minimising the expression for the mean of the forward and backward prediction error powers in (12) with respect to all of the coefficients, $a_{p1}, a_{p2}, ..., a_{pp}$, rather than just a_{pp}. Algorithms which follow this strategy[23,24] tend to yield marginally better spectral estimates

than the Burg algorithm, but their solutions are computationally more expensive and are not guaranteed to be algebraically stable.

To track non–stationarities in the signal on a time scale shorter than the segment size, consecutive segments can be made to overlap, typically by half their length although in the limit the shift from one segment to the next could be reduced to a single data sample. However, in such situations it is often better to use an adaptive model such as the Kalman filter,[3] in which the values of the AR coefficients are updated on a sample–by–sample basis, with the update being proportional to the difference between the actual value of the current sample and its predicted value using the present set of coefficients. The advantage of adaptive modelling is that it can be applied to non–stationary signals without segmentation, although the disadvantages are that it is computationally more expensive than non–adaptive modelling, and the model order, once chosen, cannot be changed as it can from one segment to the next in non–adaptive modelling. Adaptive models also produce more data than they consume (i.e. p coefficients per sample compared to p coefficients per N samples for non–adaptive models) so that for some applications the sets of AR coefficients may need to be averaged, typically every N samples.

ACKNOWLEDGMENTS

This paper was written during the course of a research project funded by Oxford Instruments plc through the DTI TAPM LINK initiative. The authors would like to thank Dr. Brendan Ruck and Dr. Mark Holt at Oxford University for their valuable comments on the draft of this paper.

REFERENCES

1. Barlow J S, Methods of Analysis of Nonstationary EEGs, with Emphasis on Segmentation Techniques: A Comparative Review, *J. Clin. Neurophysiol.*, 1985, 2(3): 267–304.

2. Isaksson A, Wennberg A, Zetterberg L H, Computer Analysis of EEG Signals with Parametric Models, *Proc. IEEE*, 1981, 69(4): 451–461.

3. Skagen D W, Estimation of Running Frequency Spectra Using a Kalman Filter Algorithm, *J. Biomed. Eng.*, 1988, 10(3): 275–279.

4. Makhoul J, Linear Prediction: A Tutorial Review, *Proc. IEEE*, 1975, 63(4): 561–580.

5. Blinowska K J, Malinowski M, Non–linear and Linear Forecasting of the EEG Time Series, *Biol. Cybernet.*, 1991, 66(2): 159–165.

6. Press W H, Teukolsky S A, Vetterling W T, Flannery B P, 'Numerical Recipes in C', 2nd Edition, Cambridge University Press, 1992.

7. Andersen N, On the Calculation of Filter Coefficients for Maximum Entropy Spectral Analysis, *Geophys.*, 1974, 39(1): 69–72.

8. Andersen N, Comments on the Performance of Maximum Entropy Algorithms, *Proc. IEEE*, 1978, 66(11): 1581–1582.

9. Cadzow J A, Spectral Estimation: An Overdetermined Rational Model Equation Approach, *Proc. IEEE*, 1982, 70(9): 907–939.

10. Kay S M, Marple S L, Spectrum Analysis — A Modern Perspective, *Proc. IEEE*, 1981, 69(11): 1380–1419.

11. Harris F J, On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform, *Proc. IEEE*, 1978, 66(1): 51–83.

12. Akaike H, A New Look at the Statistical Model Identification, *IEEE Trans. Autom. Contr.*, 1974, 19(6): 716–723.

13. Parzen E, Some Recent Advances in Time Series Modelling, *IEEE Trans. Automat. Contr.*, 1974, 19(6): 723–730.

14. Akaike H, Fitting Autoregressions for Prediction, *Ann. Inst. Statist. Math.*, 1969, 21: 243–247.

15. Rissanen J, Modelling by Shortest Data Description, *Automatica*, 1978, 14(5): 465–471.

16. Rissanen J, A Predictive Least–Squares Principle, *IMA J. Math. Contr. Inform.*, 1986, 3: 211–222.

17. Konstantinides K, Threshold Bounds in SVD and a New Iterative Algorithm for Order Selection in AR Models, *IEEE Trans. Signal Processing*, 1991, 39(5): 1218–1221.

18. Duric P M, Kay S M, Order Selection of Autoregressive Models, *IEEE Trans. Signal Processing*, 1992, 40(11): 2829–2833.

19. Dickie J R, Nandi A K, On the Performance of AR Model Order Selection Methods, *Proc. Seventh Euro. Signal Processing Conf.*, Edinburgh, 1994, 1851–1854.

20. Roberts S, Tarassenko L, New Method of Automated Sleep Quantification, *Med. & Biol. Eng. & Comput.*, 1992, 30(5): 509–517.

21. Roberts S, Tarassenko L, Analysis of the Sleep EEG Using a Multilayer Network with Spatial Organisation, *IEE Proceedings–F*, 1992, 139(6): 420–425.

22. Rechtschaffen A, Kales A, 'A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects', Public Health Service, U.S. Government Printing Office, Washington D.C., 1968.

23. Marple L, A New Autoregressive Spectrum Analysis Algorithm, *IEEE Trans. Acoust., Speech, Signal Process.*, 1980, 28(4): 441–454.

24. Ulrych T J, Clayton R W, Time Series Modelling and Maximum Entropy, *Phys. Earth & Plan. Int.*, 1976, 12: 188–200.

Appendix: derivation of the recursive relationships between e_{pn} and b_{pn}

From (2):

$$\begin{aligned} e_{pn} &= s_n + \sum_{i=1}^{p} a_{pi} s_{n-i} \\ &= s_n + \sum_{i=1}^{p-1} a_{pi} s_{n-i} + a_{pp} s_{n-p} \\ &= s_n + \sum_{i=1}^{p-1} \left[a_{(p-1)i} + a_{pp} a_{(p-1)(p-i)} \right] s_{n-i} + a_{pp} s_{n-p} \\ &= s_n + \sum_{i=1}^{p-1} a_{(p-1)i} s_{n-i} + a_{pp} [s_{n-p} + \sum_{i=1}^{p-1} a_{(p-1)(p-i)} s_{n-i}] \\ &= e_{(p-1)n} + a_{pp} [s_{n-p} + \sum_{i=1}^{p-1} a_{(p-1)(p-i)} s_{n-i}] \end{aligned}$$

But: $$s_{n-p} + \sum_{i=1}^{p-1} a_{(p-1)(p-i)} s_{n-i} = s_{n-p} + \sum_{i=1}^{p-1} a_{(p-1)i} s_{n-p+i} = b_{(p-1)(n-1)}$$

So that finally: $$e_{pn} = e_{(p-1)n} + a_{pp} b_{(p-1)(n-1)}$$

It can similarly be shown that: $$b_{pn} = b_{(p-1)(n-1)} + a_{pp} e_{(p-1)n}$$

ABBREVIATED TITLE FOR USE AS A RUNNING HEADLINE:

Parametric modelling of EEG

FIGURE CAPTIONS:

Figure 1 Techniques for parametric modelling using a rational transfer function and algorithms for autoregressive parameter estimation

Figure 2 Linear prediction using a pth–order autoregressive model

Figure 3 The Levinson–Durbin algorithm

Figure 4 Forward and backward linear prediction

Figure 5 The Burg algorithm

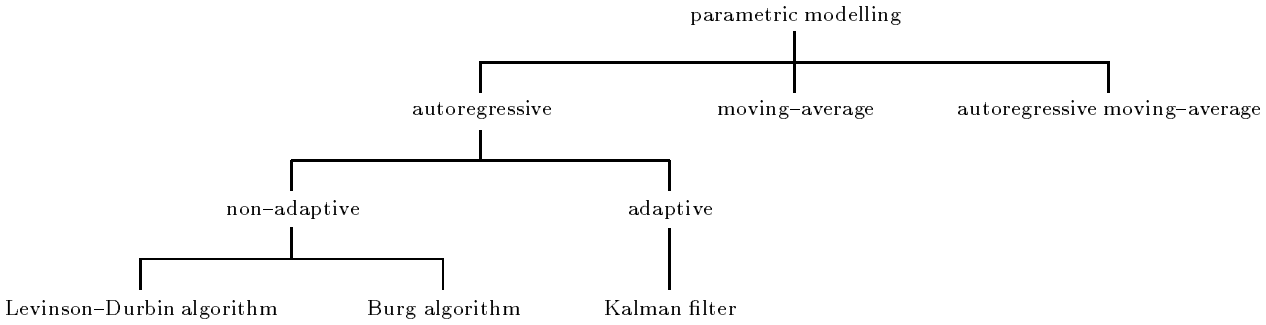
Figure 6 The interpretation of an autoregressive model as an all–pole filter

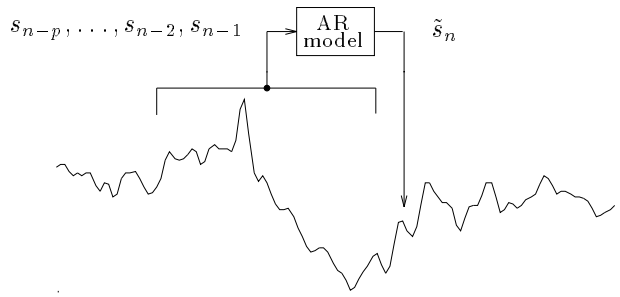
Figure 7 A comparison of spectral estimation methods

Figure 8 The mean values of E_p (solid lines) and corresponding values of the FPE (circles) for 4,800 seconds each of wakefulness, REM, and deep sleep

Figure 9 The distribution of optimum model order on a second–by–second basis

Figure 10 The variation of optimum model order with sleep stage for a 7.5 hour EEG recording


parametric modelling
autoregressive
moving–average
autoregressive moving–average
non–adaptive
adaptive
Levinson–Durbin algorithm
Burg algorithm
Kalman filter


$s_{n-p}, \ldots, s_{n-2}, s_{n-1}$
AR model
$\tilde{s}_n$

Initialisation :

$$E_0 = \tilde{R}_0$$

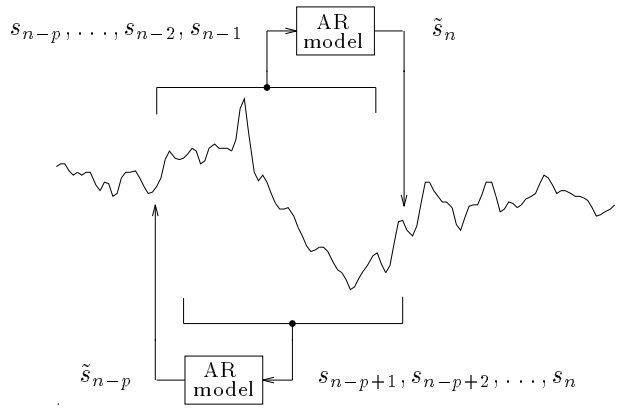
For $m = 1, 2, ..., p$:

$$k_m = -\left[\tilde{R}_m + \sum_{i=1}^{m-1} a_{(m-1)i}\tilde{R}_{m-i}\right] \Big/ E_{m-1}$$

$$a_{mm} = k_m$$

$$a_{mi} = a_{(m-1)i} + a_{mm}a_{(m-1)(m-i)}, \quad \text{for } 1 \leq i \leq m-1$$

$$E_m = (1 - k_m^2)E_{m-1}$$


$s_{n-p}, \ldots, s_{n-2}, s_{n-1}$
AR model
$\tilde{s}_n$
$\tilde{s}_{n-p}$
AR model
$s_{n-p+1}, s_{n-p+2}, \ldots, s_n$

Initialisation :

$$E_0 = \frac{1}{N}\sum_{n=1}^{N} s_n^2$$

$$e_{0n} = b_{0n} = s_n, \quad \text{for } 1 \le n \le N$$

For $m = 1, 2, ..., p$:

$$k_m = -2 \sum_{n=m+1}^{N} b_{(m-1)(n-1)} e_{(m-1)n} \Bigg/ \sum_{n=m+1}^{N} [b^2_{(m-1)(n-1)} + e^2_{(m-1)n}]$$

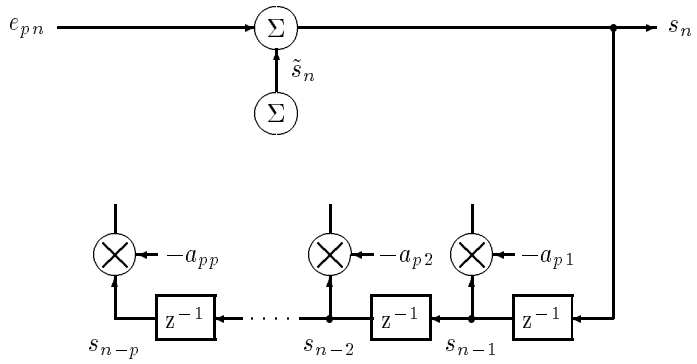
$$a_{mm} = k_m$$

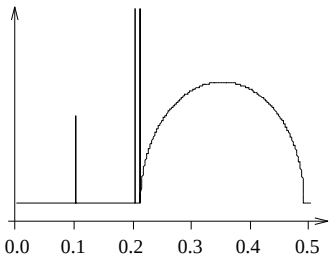
$$a_{mi} = a_{(m-1)i} + a_{mm} a_{(m-1)(m-i)}, \quad \text{for } 1 \le i \le m-1$$

$$E_m = (1 - k_m^2) E_{m-1}$$

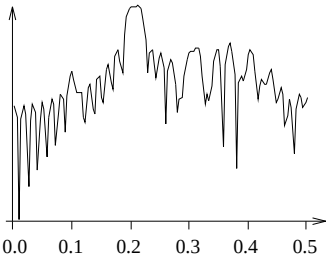
$$e_{mn} = e_{(m-1)n} + a_{mm} b_{(m-1)(n-1)}, \quad \text{for } 1 \le n \le N-m$$

$$b_{mn} = b_{(m-1)(n-1)} + a_{mm} e_{(m-1)n}, \quad \text{for } 1 \le n \le N-m$$

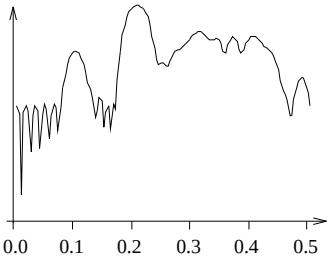

e_{pn}
Σ
s_n
$\tilde{s}_n$
Σ
$-a_{pp}$
$-a_{p2}$
$-a_{p1}$
z^{-1}
z^{-1}
z^{-1}
s_{n-p}
s_{n-2}
s_{n-1}



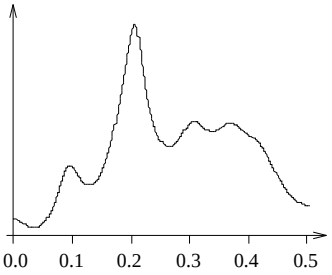
(a) ideal spectrum



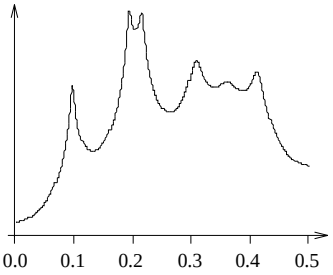
(b) periodogram method



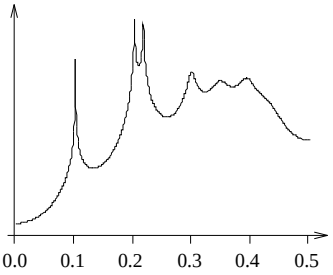
(c) periodogram method with Hamming window



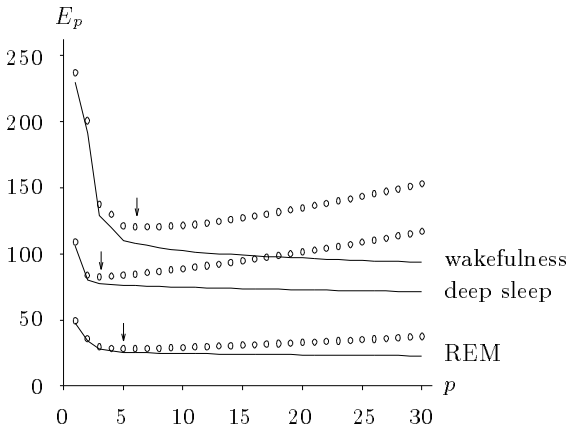
(d) autocorrelation method

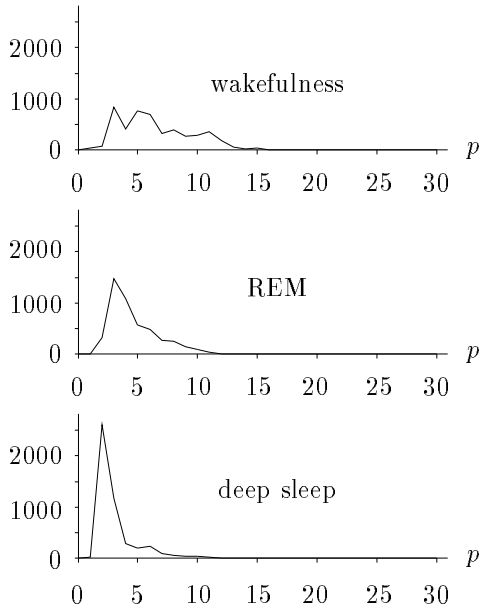


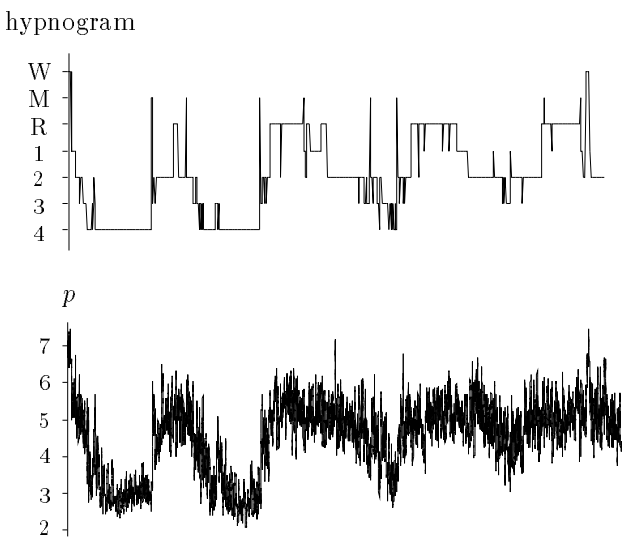
(e) autocorrelation method with Hamming window



(f) maximum entropy method


E_p
250
200
150
100
50
0
0
5
10
15
20
25
30
p
wakefulness
deep sleep
REM


2000
1000
0
wakefulness
p
0 5 10 15 20 25 30
2000
1000
0
REM
p
0 5 10 15 20 25 30
2000
1000
0
deep sleep
p
0 5 10 15 20 25 30


hypnogram
W
M
R
1
2
3
4
p
7
6
5
4
3
2