

FAST LEAST SQUARES ALGORITHMS IN LINEAR IDENTIFICATION

Paolo CARBONE, Claudio NARDUZZI, Dario PETRI, and Faabio ZANIN

Dipartimento di Elettronica e Informatica
Università di Padova, Italy
Via G. Gradenigo 6/a, I-35131 Padova, Italy

Received: July 3, 1992.

Abstract

The paper deals with the identification of FIR linear systems by time-domain least squares methods. Fast algorithms for solving the least squares problem are introduced, based on the notion of quasi-Toeplitz matrices. The estimation problem is solved by embedding it into a linear prediction one, and it is shown that the algorithms also allow the efficient solution of constrained least squares problems in a very common case. The iterative approach to constrained least squares identification is briefly considered, followed by the presentation of the applications considered by the authors. Finally, a few comments are made about the performances of the methods discussed.

Keywords: least squares, identification, fast algorithms.

Introduction

A large number of signal processing problems is solved by means of statistical signal processing algorithms. Particularly, the method of minimum variance estimation is often applied in solving linear prediction and system identification problems. A more general approach is based on the principle of maximum likelihood.

In some cases, for instance, when dealing with transient signals, such methods may be difficult to pursue. Statistical descriptions of the signals, considered as random processes, could possibly not be known, or might be difficult to infer from experimental observations. In such cases a non-stochastic approach to signal processing may be preferable.

The deterministic counterpart of the criterion of minimum variance is the well-known principle of least squares. The algorithms that can be obtained by applying the two criteria exhibit some striking similarities, although different properties exist in the stochastic and in the deterministic case.

In the identification of linear systems from noisy data, taking disturbances into account usually helps to improve accuracy; often, it is the only way by which a reasonable solution can be obtained. One possible approach

is then to recast the original problem as a constrained least squares one; its solution will depend on the values of so-called regularisation parameters. Optimization of these parameters can be obtained by an iterative procedure, in which case the availability of a fast algorithm is very important to ensure computational efficiency.

The purpose of this paper is to discuss the use of recursive least squares algorithms in the identification of linear systems. The application to constrained least squares will be emphasised, showing how fast algorithms can be employed in the iterative solution of constrained problems.

Finally, their application in some practical cases will be briefly considered.

Fast Least Squares

The problem of identifying a linear system from discrete-time input/output measurements can be presented in different ways. In the time domain, identification of a finite duration impulse response could be stated as follows: given the two sequences $x(nT)$ and $y(nT)$ of samples of the input and output signals, with n an integer, find the sequence $h(mT)$, $0 \leq m \leq M$, such that:

$$y(nT) = \sum_{m=0}^M h(mT)x[(n-m)T]. \quad (1)$$

The range of variation for n has been deliberately left undefined, but obviously it will be limited in all practical cases. Implicitly, it has been assumed that the sampling interval is the same for the two sequences, and, for the sake of simplicity, T will be henceforth set to 1. Furthermore, samples must be aligned in time: the absence of timing skews is an important but not so obvious requirement for correct identification.

Considering the solution of (1) for different values of n results in a set of linear equations which in most cases are overdetermined, requiring a least squares solution. Applying the orthogonality principle, and using a vector-matrix notation, the estimate $\mathbf{h} = [h(0), \dots, h(M)]^T$ can be shown to be the solution of the equation:

$$\mathbf{r}^{yx} = \mathbf{R}^{xx} \mathbf{h}, \quad (2)$$

where \mathbf{r}^{yx} is an $(M+1)$ -dimensional vector whose elements are defined by:

$$r^{yx}(j) = \sum_k y(k)x(k-j) \quad j = 0, 1, \dots, M, \quad (3)$$

while the elements of the $(M + 1) \times (M + 1)$ matrix \mathbf{R}^{xx} are given by:

$$r^{xx}(j, i) = \sum_k x(k - i)x(k - j) \quad j, i = 0, 1, \dots, M. \quad (4)$$

\mathbf{R}^{xx} can be called the data autocorrelation matrix of the sequence $x(n)$, while it is easily shown that \mathbf{r}^{yx} is the first column of the data cross-correlation matrix between $y(n)$ and $x(n)$.

It can be noticed that, if $-\infty < k < +\infty$, and assuming $x(n)$ and $y(n)$ to be the samples of a realisation of an ergodic process, Eqs. (3) and (4) define stochastic correlation matrices. In this case the actual impulse response could be obtained; furthermore, the autocorrelation (or covariance, if the process x has zero mean) matrix has the important structural property of being Toeplitz. This feature is the basis for the well known and efficient Levinson-Durbin recursive algorithm for the solution of identification and linear prediction problems.

In a least squares setting, the finite range of variation of the index k in expressions (3) and (4) must be specified for expression (2) to be meaningful. Usually, the sequences $x(n)$ and $y(n)$ are assumed to have the same duration, i.e., $S \leq n \leq N$ for both; several possibilities can be considered for the range of k , which result in different properties of the matrix \mathbf{R}^{xx} .

If k is allowed to vary between $S + M$ and N , each element of \mathbf{R}^{xx} results from the sum of the same number of product terms $x(k - i)x(k - j)$. No hypotheses are required on the behaviour of the two data sequences outside the interval $S \leq n \leq N$, a situation which is referred to as unwindowed data, or, somewhat improperly, covariance case. The samples of $y(n)$ for $S \leq n \leq S + M - 1$ are not employed in the estimation of \mathbf{h} . Unfortunately, in this case the matrix \mathbf{R}^{xx} is not Toeplitz.

Very often, the assumption is made that $S \leq k \leq N + M$, and that $x(n)$ and $y(n)$ are zero outside the interval $S \leq n \leq N$: the reason for this choice lies in the fact that the resulting matrix \mathbf{R}^{xx} preserves the Toeplitz structure and enables the use of efficient algorithms. However, a hypothesis has been made about the signals, which is often unrealistic and may affect the accuracy in the estimation of \mathbf{h} . In fact, ideally infinite data sequences have been supposed to be zero, i.e., windowed, outside the observation interval. Extensive studies can be found in the literature on the subject of the choice of appropriate windowing schemes.

Finally, further two possibilities exist, one being to assume that $S \leq k \leq N$, which requires $x(n) = 0$ for $n < S$ and is called the pre-windowed case; the other is $S + M \leq k \leq N + M$, for which $x(n) = 0, y(n) = 0$ for $n > N$ must be assumed, resulting in post-windowed data sequences. In

both cases the resulting matrix \mathbf{R}^{xx} is non-Toeplitz. It could be observed that the processing of transient signals, which usually start from a constant steady state, can be realised in pre-windowed form without introducing any truncation error.

In the following discussion it will be supposed that data sequences begin at time 0, hence $S = 0$. To evidence the dependence on the length of the sequences the final index N will be indicated in parentheses, e.g. $\mathbf{R}^{xx}(N)$.

The identification problem represented by Eq. (2) can be solved recursively by different computing schemes, namely, order recursions, time recursions, and time and order recursions. The former is the typical Levinson recursion, while time recursions are useful for on line identification purposes, or for adaptive filtering applications.

The first examples of fast recursive least squares algorithms were developed about fifteen years ago (FRIEDLANDER et al., 1978, LJUNG et al., 1978). They originate from the consideration that in practice the matrix \mathbf{R}^{xx} , although not Toeplitz, is in some way close to it. It was thought that the structure of a quasi-Toeplitz matrix could be exploited to obtain algorithms for matrix inversion, the solution of linear prediction problems, the computation of Kalman gains, etc., which would resemble those for Toeplitz matrices and prove nearly as efficient.

Given the matrix \mathbf{R}^{xx} , the $M \times M$ matrix $\delta\mathbf{R}^{xx}$ is defined as follows:

$$\delta(\mathbf{R}^{xx}) = [r^{xx}(j, i) - r^{xx}(j + 1, i + 1)] \quad i, j = 0, 1, \dots, M - 1. \quad (5)$$

It can be seen that $\delta(\mathbf{R}^{xx})$ is obtained as the difference between the two $M \times M$ principal minors of \mathbf{R}^{xx} . If \mathbf{R}^{xx} is Toeplitz, then $\delta(\mathbf{R}^{xx})$ is a null matrix, hence its rank is 0; for a generic matrix \mathbf{A} , $\delta(\mathbf{A})$ may have any form and its rank can be at most M . The rank of this matrix, called the displacement matrix, is taken as an indication of the closeness to the Toeplitz case.

In all the cases considered above for linear estimation the following conditions hold:

- i) $\delta(\mathbf{R}^{xx})$ is symmetric;
- ii) the rank P of $\delta(\mathbf{R}^{xx})$ is low;
- iii) $\delta(\mathbf{R}^{xx})$ can be factorised in the form:

$$\delta\mathbf{R}^{xx} = \mathbf{D}\Sigma\mathbf{D}^T, \quad (6)$$

where \mathbf{D} is an $M \times P$ matrix and Σ is a $P \times P$ signature matrix (i.e., its elements are either 1, 0 or -1), with $P \leq M$.

For instance, the correlation matrix obtained with unwindowed data generates the following displacement matrix:

$$\delta(\mathbf{R}^{xx}(N)) = \left[\sum_{k=M}^N x(k-i)x(k-j) - \sum_{k=M}^N x(k-i-1)x(k-j-1) \right]_{j,i}, \quad (7)$$

from which, in a straight forward manner, one gets:

$$\delta(\mathbf{R}^{xx}(N)) = \left[x(N-i)x(N-j) - x(M-1-i)x(M-1-j) \right]_{j,i}. \quad (8)$$

Defining the two M -dimensional vectors $\tilde{\mathbf{x}}(N)$ and $\tilde{\mathbf{x}}(M-1)$ as:

$$\begin{aligned} \tilde{\mathbf{x}}^T(N) &= \left[x(N)x(N-1)\dots x(N-M+1) \right] \\ \tilde{\mathbf{x}}^T(M-1) &= \left[x(M-1)x(M-2)\dots x(0) \right], \end{aligned} \quad (9)$$

the $M \times M$ matrix $\delta(\mathbf{R}^{xx}(N))$ can be written as follows:

$$\delta(\mathbf{R}^{xx}(N)) = \left[\tilde{\mathbf{x}}(N); \tilde{\mathbf{x}}(M-1) \right] \cdot \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \left[\tilde{\mathbf{x}}(N); \tilde{\mathbf{x}}(M-1) \right]^T, \quad (10)$$

which is exactly the factorised form given in Eq. (6). This property is important, since it allows to find some useful partitions of \mathbf{R}^{xx} and time update their expressions. Among the possible definitions of \mathbf{R}^{xx} considered above, that for the unwinded case has the displacement matrix with the highest rank, being $P = 2$.

The problem that is considered in this paper is the identification of a finite impulse response (FIR) linear system by a sequential least squares approach. In this case the system order M is given, so that the estimated impulse response is represented by the $(M+1)$ -element vector \mathbf{h} . The estimate of \mathbf{h} must be updated each time new samples of the two sequences x and y are acquired; the aim is to find a recursive procedure to obtain $\mathbf{h}(N+1)$ given $\mathbf{h}(N)$. Obviously, $\mathbf{h}(N+1)$ must satisfy the time-updated least squares equation:

$$\mathbf{r}^{yx}(N+1) = \mathbf{R}^{xx}(N+1)\mathbf{h}(N+1), \quad (11)$$

and it can be seen, by considering how the elements of \mathbf{R}^{xx} and \mathbf{r}^{yx} are defined, that:

$$\mathbf{R}^{xx}(N+1)\mathbf{h}(N) + \begin{bmatrix} x(N+1) \\ \vdots \\ \tilde{\mathbf{x}}(N) \end{bmatrix} \left(y(N+1) - [x(N+1); \tilde{\mathbf{x}}^T(N)]\mathbf{h}(N) \right) =$$

$$\mathbf{r}^{yz}(N + 1). \tag{12}$$

An updating relationship for \mathbf{h} may be found by introducing the vector $\mathbf{k}(N + 1)$, that is

$$\mathbf{R}^{zx}(N + 1)\mathbf{k}(N + 1) = [x(N + 1); \bar{\mathbf{x}}^T(N)]^T, \tag{13}$$

thus:

$$\mathbf{h}(N + 1) = \mathbf{h}(N) + \mathbf{k}(N + 1)(y(N + 1) - [x(N + 1); \bar{\mathbf{x}}^T(N)]\mathbf{h}(N)). \tag{14}$$

A recursive algorithm can be obtained if indications are given as to how the vector \mathbf{k} is updated. It should be noticed that, according to its definition, \mathbf{k} is equivalent to the Kalman gain for the matrix \mathbf{R}^{zx} .

The problem of the iterative calculation of \mathbf{k} can be embedded into the recursive solution of a linear prediction problem, i.e., computing the one-step forward and backward linear predictors of order M , given the data sequence $x(n)$ (MARPLE, 1981). Let the forward predictor be defined by the $M \times 1$ coefficient vector $\mathbf{a}=[a(1) \dots a(M)]^T$ and the backward predictor by $\mathbf{b}=[b(M) \dots b(1)]^T$; then, assuming:

$$\alpha(N) = [1; -\mathbf{a}^T(N)]^T, \quad \beta(N) = [-\mathbf{b}^T(N); 1]^T, \tag{15}$$

one has:

$$\mathbf{R}^{zx}(N)\alpha(N) = [\sigma^f(N) 0 \dots 0]^T, \quad \mathbf{R}^{zx}(N)\beta(N) = [0 \dots 0 \sigma^b(N)]^T, \tag{16}$$

where σ^f and σ^b are the sums of squared forward and backward prediction errors up to time N .

It is interesting to note that \mathbf{R}^{zx} can always be partitioned as follows:

$$\mathbf{R}^{zx} = \begin{bmatrix} \mathbf{R}^b & \vdots & \mathbf{r}^b \\ \dots & \vdots & \dots \\ \mathbf{r}^{bT} & \vdots & * \end{bmatrix} = \begin{bmatrix} * & \vdots & \mathbf{r}^{fT} \\ \dots & \vdots & \dots \\ \mathbf{r}^f & \vdots & \mathbf{R}^f \end{bmatrix}, \tag{17}$$

where $*$ denotes a scalar, \mathbf{R}^b and \mathbf{R}^f are $M \times M$ matrices and the following relationships hold:

$$\mathbf{r}^f = \mathbf{R}^f \mathbf{a}, \quad \mathbf{r}^b = \mathbf{R}^b \mathbf{b}. \tag{18}$$

Since \mathbf{R}^b and \mathbf{R}^f are the two principal minors of \mathbf{R}^{zx} one may also write $\delta(\mathbf{R}^{zx}) = \mathbf{R}^b - \mathbf{R}^f$; if \mathbf{R}^{zx} is a Toeplitz data correlation matrix $\delta(\mathbf{R}^{zx}) = 0$,

hence $\mathbf{R}^b = \mathbf{R}^f$ and, since $\mathbf{r}^b = \mathbf{r}^f$, the forward and backward predictor must coincide, which is a well known fact.

When the new sample $x(N+1)$ is acquired, the updated displacement matrix satisfies the equality:

$$\sigma(\mathbf{R}^{xx}(N+1)) = \begin{bmatrix} \tilde{\mathbf{x}}(N+1) & \tilde{\mathbf{x}}(M-1) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} \tilde{\mathbf{x}}(N+1) & \tilde{\mathbf{x}}(M-1) \end{bmatrix}^T, \quad (19)$$

and, combining (10) with (17) and (19), some useful relationships can be obtained:

$$\begin{aligned} \mathbf{R}^f(N+1) &= \mathbf{R}^f(N) + \tilde{\mathbf{x}}(N)\tilde{\mathbf{x}}^T(N), \\ \mathbf{R}^f(N+1) &= \mathbf{R}^b(N) + \tilde{\mathbf{x}}(M-1)\tilde{\mathbf{x}}^T(M-1), \\ \mathbf{R}^b(N+1) &= \mathbf{R}^b(N) + \tilde{\mathbf{x}}(N+1)\tilde{\mathbf{x}}^T(N+1). \end{aligned} \quad (20)$$

To carry out the updatings, three auxiliary $M \times 1$ vectors \mathbf{c} , \mathbf{d} , and \mathbf{e} need to be introduced and defined as follows:

$$\begin{aligned} \mathbf{R}^f(N+1)\mathbf{c}(N+1) &= \tilde{\mathbf{x}}(N), \\ \mathbf{R}^f(N+1)\mathbf{d}(N) &= \tilde{\mathbf{x}}(M-1), \\ \mathbf{R}^f(N+1)\mathbf{e}(N+1) &= \tilde{\mathbf{x}}(N+1). \end{aligned} \quad (21)$$

A full derivation of the recursive algorithm for the unwindowed case is presented in (HALKIAS et al., 1982). Its steps are summarised here:

1) compute the forward prediction error e before updating the predictor:

$$e^f(N+1) = x(N+1) - a^T(N)\tilde{\mathbf{x}}(N),$$

2) update the forward predictor:

$$\mathbf{a}(N+1) = \mathbf{a}(N) + \mathbf{c}(N+1)e^f(N+1),$$

3) compute the updated forward prediction error ε^f :

$$\varepsilon^f(N+1) = x(N+1) - \mathbf{a}^T(N+1)\tilde{\mathbf{x}}(N),$$

4) compute the sum:

$$\sigma^f(N+1) = \sigma^f(N) + \varepsilon^f(N+1)e^f(N+1),$$

5) update the vector \mathbf{k} :

$$\mathbf{k}(N+1) = \begin{bmatrix} \dots & 0 & \dots \\ \dots & c(N+1) & \dots \end{bmatrix} + \frac{\varepsilon^f(N+1)}{\sigma^f(N+1)} \mathbf{a}(N+1),$$

6) update the estimated impulse response according to Eg . (14).

For the auxiliary terms the following recursions are needed:

7) compute the backward prediction error e^b before updating the predictor:

$$e^b(N+1) = x(N-M+1) - \mathbf{b}^T(N)\tilde{\mathbf{x}}(N+1),$$

8) partition the vector \mathbf{k} :

$$\mathbf{k}(N+1) = \begin{bmatrix} \mathbf{m}(N+1) \\ \mu(N+1) \end{bmatrix},$$

9) update \mathbf{e} :

$$\mathbf{e}(N+1) = \frac{\mathbf{m}(N+1) + \mu(N+1)\mathbf{b}(N)}{1 - e^b(N+1)\mu(N+1)},$$

10) update the backward predictor \mathbf{b} :

$$\mathbf{b}(N+1) = \mathbf{b}(N) + \mathbf{e}(N+1)e^b(N+1),$$

11) update \mathbf{c} :

$$\mathbf{c}(N+2) = \frac{\mathbf{e}(N+1) - \left[\tilde{\mathbf{x}}^T(M-1)\mathbf{e}(N) \right] \mathbf{d}(N)}{1 - \left[\tilde{\mathbf{x}}^T(M-1)\mathbf{e}(N+1) \right] \left[\tilde{\mathbf{x}}^T(N)\mathbf{d}(N) \right]}.$$

It should be noticed from the definition of \mathbf{c} given in (21) that computing this vector requires all data samples except the most recent one. This explains why it can actually be determined in advance.

12) update \mathbf{d} :

$$\mathbf{d}(N+1) = \mathbf{d}(N) - \left[\tilde{\mathbf{x}}^T(N)\mathbf{d}(N) \right] \mathbf{c}(N+2).$$

The most important feature of this algorithm is that the number of operations per step is proportional to M , whereas conventional recursive least squares algorithms require a number of operations proportional to M^2 .

The algorithm must be initialised properly to obtain correct results. This aspect will be discussed in the next section, after introducing constrained least squares problems.

Constrained Least Squares

The identification of a linear system can be obtained by processing the results of an input/output experiment. The relevant data sequences must be acquired by a sampling instrument, and processed according to Eq. (2) to obtain an estimate of the FIR impulse response \mathbf{h} . Unfortunately, uncertainties and errors are introduced, both in the measurement process and in modelling the linear system; to reflect this, the input and output data sequences should be rewritten in the form $x(n) + e_x(n)$ and $y(n) + e_y(n)$, respectively. The presence of uncertainties can be expected to affect the accuracy of the estimate.

Very often, it is assumed that input uncertainties, represented by $e_x(n)$, are negligible, so that only the term $e_y(n)$ is taken into account. From (2) one gets the following estimate:

$$\mathbf{h}(N) = (\mathbf{R}^{xx}(N))^{-1}(\mathbf{r}^{yx}(N) + \mathbf{r}^{ex}(N)), \quad (22)$$

where in the covariance case $\mathbf{R}^{xx}(N)$ and $\mathbf{r}^{yx}(N)$ are defined according to (3) and (4) with $M \leq k \leq N$ and, similarly, the elements of $\mathbf{r}^{ex}(N)$ satisfy:

$$r^{ex}(j) = \sum_{k=M}^N e_y(k)x(k-j) \quad j = 0, 1, \dots, M. \quad (23)$$

The inverse solution of a convolution integral is known to be an ill-posed problem; consequently, if data are acquired from noisy signals, or uncertainties are present, the estimate given by (22) can be expected to be inaccurate, owing to the effect of the term \mathbf{r}^{ex} . Often, an excessively noisy or oscillating solution is found.

A classical approach to this problem is to introduce bounds on the solution, which typically means to limit the maximum energy of \mathbf{h} , i.e., $\mathbf{h}^T \mathbf{h} \leq E$, with $E > 0$ and real, or of some derivative of it. In general, expressing by the matrix \mathbf{C} a linear operator on \mathbf{h} , the bound takes the form:

$$\mathbf{h}^T \mathbf{C}^T \mathbf{C} \mathbf{h} \leq E. \quad (24)$$

The solution of this constrained least squares problem is quite straightforward and can be obtained, for instance, by using Lagrange multipliers:

$$\mathbf{h}(N) = (\mathbf{R}^{xx}(N) + \gamma \mathbf{C}^T \mathbf{C})^{-1}(\mathbf{r}^{yx}(N) + \mathbf{r}^{ex}(N)), \quad (25)$$

where γ is the inverse of the Lagrange multiplier. For the problem to be completely solved, the optimum value of this parameter must be found.

Unfortunately, expressing γ in closed form is nearly impossible, at least in a deterministic approach. One way to circumvent this difficulty is to obtain γ by iteratively solving the constrained least squares problem until an optimal, or at least suboptimal estimate of \mathbf{h} is found (TWOMEY, 1965, HUNT, 1971).

Such a procedure may be long and computationally demanding, since several iterations could be necessary. Therefore, it is particularly useful in this case to investigate the applicability of fast recursive least squares algorithms. Fortunately, in the most common constrained case all that is required is an appropriate initialisation of the algorithm already discussed.

Considering that before data are gathered all sums in the defining equation (4) yield zero, it is necessary to avoid starting with a singular matrix. The simplest and most common way to do this is to assume $\mathbf{R}^{xx}(M-1) = \delta \mathbf{I}$, where \mathbf{I} is an $(M+1) \times (M+1)$ identity matrix, and δ is a scalar. To obtain this, the algorithm should be initialised as follows:

$$\mathbf{a}(M-1) = \mathbf{b}(M-1) = 0,$$

$$\sigma^f(M-1) = \delta,$$

$$\mathbf{d}(M-1) = \mathbf{c}(M) = \tilde{\mathbf{x}}(M-1) \left[\delta + \tilde{\mathbf{x}}^T(M-1) \tilde{\mathbf{x}}(M-1) \right]^{-1}.$$

It should be noticed that in fast algorithms the data correlation matrix \mathbf{R}^{xx} is never calculated explicitly. Rather, it is implicitly defined by Eq. (13), that is, by the expression for \mathbf{k} . Since this vector is updated by using the recursive relationships (20), it suffices to determine \mathbf{R}^{xx} at any one time step to verify its expression.

Using the recursions given in the previous section, it is relatively simple to obtain the expression of $\mathbf{k}(M)$:

$$\mathbf{k}(M) = \begin{bmatrix} x(M) \\ \tilde{\mathbf{x}}(M-1) \end{bmatrix} \left(\sigma^f(M) \right)^{-1} = \begin{bmatrix} x(M) \\ \tilde{\mathbf{x}}(M-1) \end{bmatrix} \left(\delta + [x(M); \tilde{\mathbf{x}}^T(M-1)]^T \begin{bmatrix} x(M) \\ \tilde{\mathbf{x}}(M-1) \end{bmatrix} \right)^{-1}. \quad (26)$$

Further, by applying the matrix inversion lemma, and making use of the matrix identity $\mathbf{I} - \mathbf{B}(\mathbf{A} + \mathbf{B})^{-1} = \mathbf{A}(\mathbf{A} + \mathbf{B})^{-1}$, one gets:

$$\mathbf{k}(M) = \begin{bmatrix} x(M) \\ \tilde{\mathbf{x}}(M-1) \end{bmatrix} \left[[x(M); \tilde{\mathbf{x}}^T(M-1)]^T + \delta \mathbf{I} \right]^{-1} \begin{bmatrix} x(M) \\ \tilde{\mathbf{x}}(M-1) \end{bmatrix}. \quad (27)$$

This equation shows that the matrix implicitly defined by $\mathbf{k}(M)$ is in fact $(\mathbf{R}^{xx}(M) + \delta\mathbf{I})$. If a straight forward least squares problem is to be solved, δ should be kept as small as possible, compatibly with the numerical stability of the algorithm.

However, the same matrix also appears in the most common form of constrained estimate of \mathbf{h} , namely when the energy of the estimate is bounded to be $\mathbf{h}^T\mathbf{h} \geq E$, so that the matrix \mathbf{C} in Eq. (25) becomes an identity. This implies that the least squares algorithm detailed in the previous section allows the fast solution of the problem, provided $\delta = \gamma$.

The availability of an algorithm that requires $O(M)$ operations per step, instead of $O(M^2)$ as traditional iterative algorithms do, makes the iterative approach particularly attractive.

To develop an effective mean for solving identification problems, a criterion must also be given to find the optimum value of the parameter γ . This proves to be quite difficult, since very often, even though the condition $\mathbf{h}^T\mathbf{h} \leq E$ is specified, no actual value can be given for the constant E . On the other hand, the least squares criterion tries to minimise the trace of the data correlation matrix for the output reconstruction error, but this condition seldom corresponds with the most accurate estimate of \mathbf{h} . Unfortunately, a general criterion has not been found so far; several approaches have been proposed, depending on the nature of the system under analysis and of the test signals used to identify it.

When the system is band-limited and step-like waveforms are used to test it, good results can be obtained by imposing that the output mean squared error is uniform at all frequencies. Step-like waveforms allow to differentiate very easily between portions of the signal which are nearly constant and parts where abrupt variations of the signal produce broad-band spectral contributions. It is therefore very simple, even in the time domain, to calculate two distinct error values, a low-frequency and a high-frequency one, compare them and change the value of γ until the optimal condition is found where the two approximately coincide.

This method provides a condition that is very easily detected; results obtained so far, and presented in (BERTOCCO et al., 1991), showed that it performs well in many conditions.

Applications

There are many possibilities to employ linear identification algorithms. The time domain least squares approach that has been presented in this paper originated from the need to identify linear systems from input/output experiments, using broad-band transient signals. Although frequency analysis

using sinusoidal waveforms has long been standard practice in electronic engineering, it is not always possible to characterise linear systems in this way. A simple alternative is the analysis of the step response which, in practice, is often the response to a step-like waveform. Particular care in processing measured data is needed if an acceptable degree of accuracy is to be obtained. For instance, even with very clean signals the presence of quantisation noise may require attention and suggest the use of a constrained least squares procedure.

Fast algorithms have been used for the identification of a number of linear systems. So far, it has been shown that the frequency response obtained from the Fourier transform of the estimated impulse response differs from the measured reference response for no more than ± 1 dB in amplitude and ± 10 degrees in phase. Some of the results have been presented in (NARDUZZI *et al.*, 1991).

One of the limits of the method is that at present the fast algorithm can be applied to just one of the possible linear constraints, that is, the case when $\mathbf{h}^T \mathbf{h} \leq E$, and $\mathbf{C} = \mathbf{I}$. In step response tests, this kind of constraint tends to emphasize the low pass nature of the system under test, thus distorting the results. It is expected that by choosing more appropriate constraining operators the accuracy of results can be improved further, and it seems reasonable to predict that it will be possible to determine a system response to $\pm 1\%$ within its passband.

The low-pass emphasis affects adversely the attempts to synthesize compensating filters for bandwidth enhancement of low-pass systems. It is quite simple to reverse the roles of the input and output data sequence and employ the identification algorithm to obtain an inverse filter. Constrained estimation is essential in this case, since signal components within the system dead band, or even the transition band, are more strongly affected by noise. However, because of the low-pass effect of the constraint, performance of the inverse filter is somewhat limited. An attempt has been made to employ this technique for the compensation of high-voltage measurement data acquired through a damped capacitive voltage divider (NARDUZZI *et al.*, 1989). Results, although encouraging, are not yet up to the required accuracy.

At present, the research on fast algorithms is continuing, concerning in particular the possibility to obtain different kinds of linearly constrained estimate. It is hoped that, under suitable hypotheses, a proper initialisation of the algorithm variables could still enable to achieve this goal.

References

- BERTOCCO, M. – NARDUZZI, C. – OFFELLI, C. – PETRI, D. (1991): An Improved Method for Iterative Identification of Bandlimited Linear Systems. *IEEE Instrumentation and Meas. Technology Conf.* May 1991, pp. 368-372.
- CARAYANNIS, G. – DOLOGLOU, J. – EMMANOPOULOS, D. – HALKIAS, C.C. (1982): A New Generalized Recursion for the Fast Computation of the Kalman Gain to Solve the Covariance Equations. *Proc. ICASSP* 1982, pp. 1760-1763.
- FALCONER, D. – LJUNG, L. – MORF, M. (1978): Fast Calculations of Gain Matrices for Recursive Estimation Schemes. *Int.J.Control*, Vol.27, Jan.1978, pp. 1-19.
- FRIEDLANDER, B. – KAILATH, T. – LJUNG, L. T. – MORF, M. (1978): Extended Levinson and Chandrasekhar Equations for General Discrete-Time Linear Estimation Problems. *IEEE Trans. on Autom. Control*, Vol. AC-23, No.4, Aug. 1978, pp. 653-659.
- HUNT, B. R. (1971): Biased Estimation for Non Parametric Identification of Linear Systems. *Math. Biosc.*, No.10, Oct. 1971, pp. 215-237.
- MARPLE, S.L. (1981): Efficient Least Squares FIR System Identification. *IEEE Trans. Acoust. Speech and Signal Process.*, Vol. ASSP-29, No.1, Feb.1981, pp. 62-73.
- NARDUZZI, C. – ZINGALES, G. (1989): An Inverse Filtering Approach to Reconstruction of HV Impulses. *6th Int. Symp. on High Voltage Eng.*, New Orleans, LA, Aug.28 - Sept.1, 1989.
- NARDUZZI, C. – OFFELLI, C. (1991): A Time-Domain Method for the Accurate Characterization of Linear Systems. *IEEE Trans. on Instr. and Meas.*, Vol.40, No.2, April 1991, pp. 415-419.
- TWOMEY, S. (1965): The Application of Numerical Filtering to the Solution of Integral Equations Encountered in Indirect Sensing Measurements. *J.Franklin Inst.*, Vol.279, No.2, February 1965, pp. 95-109.