MULTIDIMENSIONAL SIGNAL PROCESSING

This article presents the basic concepts of multidimensional signal processing and their application to system identification. Examples of multidimensional signals include images in two dimensions and video in three dimensions. There are a number of good reference texts on multidimensional signal processing; for example, see Refs. 1, 2, and 3. One cannot discuss multidimensional signal processing without first discussing sampling in multidimensions. This concept of sampling is based on the mathematics of lattices [e.g., (4)]. The engineering analysis of sampling began in the classic paper by Petersen and Middleton (5). The multidimensional z-transform is carefully described in Ref. 6. This article covers basic problems in systems identification and presents a workable approach to their solution.

We present a systematic approach to the sampling of a multidimensional signal, which involves the mathematical concept of a sampling lattice. We then introduce multidimensional sampled signals by way of multidimensional z-transforms and multidimensional Fourier transforms. In the next section, we present an introduction to an exciting new application area of multidimensional signal processing—system identification. Then we present examples and applications of system identification. Finally, we offer concluding remarks.

SAMPLING LATTICES

There are many ways to choose the sampling geometry, so sampling a multidimensional signal is more complicated than sampling a one-dimensional signal. Although sampling points can be arranged on a rectangular grid, often there are more efficient ways to sample multidimensional signals. Nonrectangular sampling can efficiently minimize the number of points needed to characterize an M-dimensional hypervolume. As Fig. 1 shows, if the functions of interest are bandlimited over a circular region, then significant savings are possible if hexagonal sampling is used instead of rectangular sampling.

In order to describe precisely and conveniently an arbitrary sampling geometry, we must appeal to the language of linear algebra using the mathematical theory of sampling lattices. The set of all k-dimensional integer vectors will be called the *fundamental lattice,* and it will be denoted $\mathcal{N}$. That is,

$$\mathcal{N} = \{\boldsymbol{r} = [\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_k]^{\mathrm{T}} | \boldsymbol{r}_i \text{ is an integer}\}$$

The set of all k-dimensional real vectors will be denoted $\mathcal{E}$.

Now let us review a few definitions related to sums of vectors. A *linear combination* of N vectors $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\} \in \mathcal{N}$ is an expression of the form

$$\sum_{i=1}^{N} n_i \boldsymbol{r}_i$$

where n_i, $i = 1, \ldots, N$, are integers and are called coefficients. The set of vectors $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}$ is said to be *linearly independent* if

$$\sum_{i=1}^{N} n_i \boldsymbol{r}_i = 0 \Rightarrow n_i = 0 \qquad \text{for all } i$$

If the set of vectors $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}$, is linearly independent, then the totality of vectors of the form

$$\left\{ \sum_{i=1}^{N} n_i \boldsymbol{r}_i | \, n_i \text{ is an integer} \right\}$$

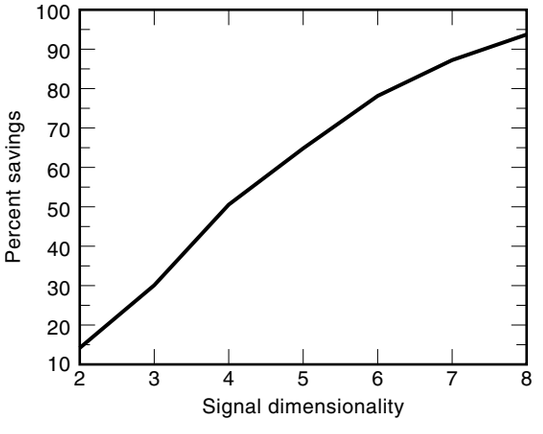

Figure 1. Percent savings: Hexagonal versus rectangular sampling.

is called a N-dimensional *lattice*, and it will be denoted by $\mathcal{R}$. In other words, the space $\mathcal{R}$ *spanned* by the set of vectors $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}$ is the space consisting of all linear combinations of the vectors. The set of vectors $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}$ is a *basis* for $\mathcal{R}$ if the vectors are linearly independent and the space spanned by $\{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}$ is equal to $\mathcal{R}$. Then we say that $\mathcal{R}$ has dimension N.

To better understand the definition of the lattice, consider the following illustration in two dimensions. Let

$$\boldsymbol{r}_1 = \begin{bmatrix} r_{11} \\ r_{21} \end{bmatrix}, \boldsymbol{r}_2 = \begin{bmatrix} r_{12} \\ r_{22} \end{bmatrix}, \quad \text{and} \quad \boldsymbol{n} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix}$$

Then

$$n_1 \boldsymbol{r}_1 + n_2 \boldsymbol{r}_2 = n_1 \begin{bmatrix} r_{11} \\ r_{21} \end{bmatrix} + n_2 \begin{bmatrix} r_{12} \\ r_{22} \end{bmatrix}$$

or equivalently,

$$n_1 \boldsymbol{r}_1 + n_2 \boldsymbol{r}_2 = \begin{bmatrix} n_1 r_{11} + n_2 r_{12} \\ n_1 r_{21} + n_2 r_{22} \end{bmatrix}$$

This can be rewritten as a matrix-vector product:

$$n_1 \boldsymbol{r}_1 + n_2 \boldsymbol{r}_2 = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} = [\boldsymbol{r}_1 \quad \boldsymbol{r}_2]\boldsymbol{n} = \boldsymbol{R}\boldsymbol{n}$$

where the matrix $\boldsymbol{R}$ is called the *sampling matrix.* In general, let $\boldsymbol{r}_i$ be the ith column of the matrix $\boldsymbol{R}$, that is,

$$\boldsymbol{R} = [\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_N]$$

then the sampling matrix $\boldsymbol{R}$ is said to generate the lattice $\mathcal{R}$. As such, the lattice $\mathcal{R}$, defined by the sampling matrix $\boldsymbol{R}$ and denoted LAT($\boldsymbol{R}$), is also given by

$$\mathcal{R} = \mathrm{LAT}(\boldsymbol{R}) = \{\boldsymbol{m} \in \mathcal{N} | \boldsymbol{m} = \boldsymbol{R}\boldsymbol{n} \text{ for } \boldsymbol{n} \in \mathcal{N}\}$$

If $\boldsymbol{R}$ is the identity matrix, then each $\boldsymbol{r}_i$ is a unit vector pointing in the ith direction, and the resulting lattice, $\mathcal{R}$, is the *fundamental lattice* $\mathcal{N}$.

Let us present some examples of sampling lattices using black dots to represent the lattice points and white circles to represent points in $\mathcal{N}$ that are not in LAT($\boldsymbol{R}$). The rectangular sampling lattice defined by the sampling matrix

$$\boldsymbol{R} = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$$

is depicted in Fig. 2. The hexagonal sampling lattice defined by the sampling matrix

$$\boldsymbol{R} = \begin{bmatrix} 1 & 1 \\ 2 & -2 \end{bmatrix}$$

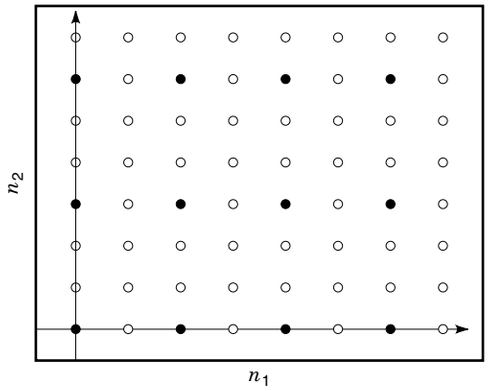

Figure 2. Lattice structure using rectangular sampling matrix.

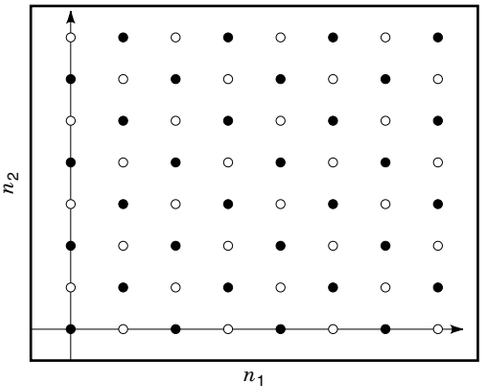

Figure 4. Lattice sampling using quincunx sampling matrix.

is depicted in Fig. 3. The quincunx sampling lattice defined by the sampling matrix

$$\boldsymbol{R} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$$

is depicted in Fig. 4.

For a given sampling matrix $\boldsymbol{R}$, the corresponding Fourier domain sampling matrix is $2\pi\boldsymbol{R}^{-\mathrm{T}}$, and the lattice it generates is called the reciprocal lattice.

The matrix that generates a lattice is not unique. As we will see later in this subsection, the following matrices generate the same lattice:

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 2 & -4 \end{bmatrix} \quad \text{and} \quad \boldsymbol{S} = \begin{bmatrix} 1 & 1 \\ 2 & -2 \end{bmatrix}$$

The theory underlying the nonuniqueness of these sampling lattices is based on unimodular matrices. Thus, in order to discuss the nonuniqueness of sampling lattices, we must first briefly discuss unimodular matrices. An integer-valued matrix $\boldsymbol{A}$ is a unimodular matrix if $|\det \boldsymbol{A}| = 1$. Unimodular matrices have many interesting properties. For example, if $\boldsymbol{A}$ is an integer-valued unimodular matrix, then $\boldsymbol{A}^{-1}$ exists and is an integer-valued unimodular matrix. In addition, given a nonsingular matrix $\boldsymbol{M}$ and an integer-valued unimodular matrix $\boldsymbol{V}$, then $\mathrm{LAT}(\boldsymbol{MV}) = \mathrm{LAT}(\boldsymbol{M})$. For notation purposes, let $\mathrm{J}(\boldsymbol{M}) = |\det \boldsymbol{M}|$, the absolute value of the determinant of the sampling matrix $\boldsymbol{M}$. It can be easily shown that given a nonsingular matrix $\boldsymbol{M}$ and an integer-valued unimodular matrix $\boldsymbol{V}$, $\mathrm{J}(\boldsymbol{M}) = \mathrm{J}(\boldsymbol{MV}) = \mathrm{J}(\boldsymbol{VM})$.

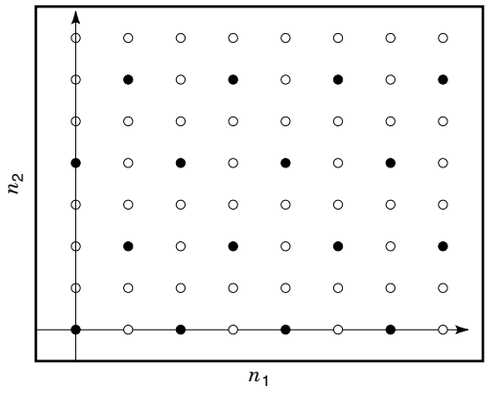

Figure 3. Lattice sampling using hexagonal sampling matrix.

Therefore $\mathrm{J}(\boldsymbol{M})$ is unique and independent of the choice of basis vectors. Moreover $\mathrm{J}(\boldsymbol{M})$ can be interpreted geometrically as the k-dimensional volume of the parallelepiped defined by $\boldsymbol{M}$. Sometimes $1/\mathrm{J}(\boldsymbol{M})$ is called the sampling density. Now, consider the following sampling matrices:

$$\boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 2 & -4 \end{bmatrix} \quad \text{and} \quad \boldsymbol{S} = \begin{bmatrix} 1 & 1 \\ 2 & -2 \end{bmatrix}$$

where $\mathrm{J}(\boldsymbol{R}) = \mathrm{J}(\boldsymbol{S}) = 4$. Since $\boldsymbol{RE} = \boldsymbol{S}$ for a unimodular matrix

$$\boldsymbol{E} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

and $\boldsymbol{R}$ and $\boldsymbol{S}$ generate the same sampling lattice. In this case the sampling lattice is known as a hexagonal sampling lattice.

Given an integer-valued matrix $\boldsymbol{R}$, a *unit cell* includes one lattice point from $\mathrm{LAT}(\boldsymbol{R})$ and $\mathrm{J}(\boldsymbol{R}) - 1$ adjacent points in $\mathcal{N}$ that are not in $\mathrm{LAT}(\boldsymbol{R})$. If these unit cells are periodically replicated on $\mathrm{LAT}(\boldsymbol{R})$, then the entire space is tiled with no overlap. Thus the unit cell is a footprint that characterizes the sampling lattice. Given an integer-valued matrix $\boldsymbol{R}$, the *fundamental parallelepiped* of lattice $\mathrm{LAT}(\boldsymbol{R})$, denoted $\mathrm{FPD}(\boldsymbol{R})$, is the unit cell that includes the origin and is bounded by all lattice points one positive unit away. Formally the fundamental parallelepiped is given by

$$\mathrm{FPD}(\boldsymbol{R}) = \{\boldsymbol{y} \in \mathcal{E} | \boldsymbol{y} = \boldsymbol{Rx} \text{ for all } \boldsymbol{x} \in [0, 1)^k\}$$

where $\mathcal{E}$ is the set of all k-dimensional real vectors.

Consider the following example as an illustration of the concept of fundamental parallelepipeds. Assume that the sampling matrix

$$\boldsymbol{R} = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$$

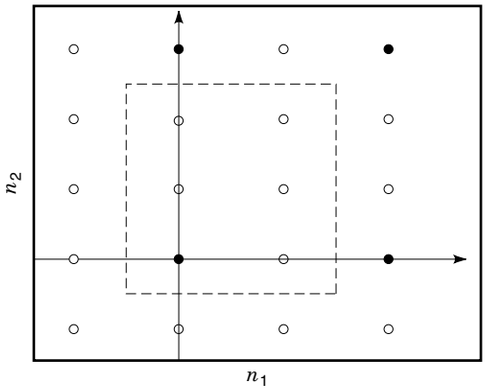

Figure 5. Fundamental parallelpiped example.

Then the fundamental parallelepiped is as given in Fig. 5. By inspection, there are $J(\boldsymbol{R}) = 6$ points in the fundamental parallelepiped.

It can be easily shown that given an integer-valued unimodular matrix $\boldsymbol{U}$ and an integer-valued diagonal matrix Λ,

$$\mathrm{FPD}(\boldsymbol{UA}) = \boldsymbol{U}\,FPD(\Lambda)$$

Sometimes in the literature, authors refer to the symmetric parallelepiped, denoted $\mathrm{SPD}(\boldsymbol{R})$. It is defined by

$$\mathrm{SPD}(\boldsymbol{R}) = \{\boldsymbol{y} \in \varepsilon | \boldsymbol{y} = \boldsymbol{Rx} \text{ for all } \boldsymbol{x} \in [-1,\ 1)^k\}$$

Let $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ be $k \times k$ integer matrices. $\mathrm{LAT}(\boldsymbol{V}_2)$ is called a *sublattice* of $\mathrm{LAT}(\boldsymbol{V}_1)$ if $\mathrm{LAT}(\boldsymbol{V}_2) \subseteq \mathrm{LAT}(\boldsymbol{V}_1)$, that is, every point of $\mathrm{LAT}(\boldsymbol{V}_2)$ is also a point of $\mathrm{LAT}(\boldsymbol{V}_1)$.

Let $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ be integer matrices, where $\mathrm{LAT}(\boldsymbol{V}_2) \subseteq \mathrm{LAT}(\boldsymbol{V}_1)$. Then, for every $\boldsymbol{m} \in \mathcal{N}$, there exist $\boldsymbol{n} \in \mathcal{N}$ such that

$$\boldsymbol{V}_1\boldsymbol{n} = \boldsymbol{V}_2\boldsymbol{m}$$

or equivalently,

$$\boldsymbol{n} = \boldsymbol{V}_1^{-1}\boldsymbol{V}_2\boldsymbol{m}$$

Since $\boldsymbol{n}$ and $\boldsymbol{m}$ are integer vectors and since $\boldsymbol{V}_1\boldsymbol{n} = \boldsymbol{V}_2\boldsymbol{m}$, then $\boldsymbol{V}_1^{-1}\boldsymbol{V}_2$ must be an integer-valued matrix. Let $\boldsymbol{L} = \boldsymbol{V}_1^{-1}\boldsymbol{V}_2$. Then $\boldsymbol{V}_1\boldsymbol{L} = \boldsymbol{V}_2$. Since $\det \boldsymbol{V}_1\boldsymbol{L} = (\det \boldsymbol{V}_1)(\det \boldsymbol{L})$,

$$\det \boldsymbol{V}_2 = (\det \boldsymbol{V}_1)(\det \boldsymbol{L})$$

so that

$$J(\boldsymbol{V}_2) = J(\boldsymbol{V}_1)(\det \boldsymbol{L})$$

Hence, $J(\boldsymbol{V}_2)$ is an integer multiple of $J(\boldsymbol{V}_1)$. For example, if

$$\boldsymbol{V}_2 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$

and

$$\boldsymbol{V}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

then $\boldsymbol{V}_2$ is a sublattice of $\boldsymbol{V}_1$. As such, $J(\boldsymbol{V}_2) = 16$ and $J(\boldsymbol{V}_1) = 4$; then $J(\boldsymbol{V}_2)$ is, as expected, an integer multiple of $J(\boldsymbol{V}_1)$, which means that $J(\boldsymbol{V}_2) = 4J(\boldsymbol{V}_1)$. An important special case results when the lattice $\mathrm{LAT}(\boldsymbol{V}_1)$ coincides with the fundamental lattice $\mathcal{N}$, namely $\mathrm{LAT}(\boldsymbol{V}_1) = \mathrm{LAT}(\boldsymbol{I})$. In addition $\rho = \boldsymbol{J}(\boldsymbol{V}_2)/\boldsymbol{J}(\boldsymbol{V}_1)$ represents the number of cells of $\mathrm{FPD}(\boldsymbol{V}_1)$ that can fit into $\mathrm{FPD}(\boldsymbol{V}_2)$. The lattice point in each one of these cells can be thought of as a shift vector $\boldsymbol{a}$, which, if added to each vector of $\mathrm{LAT}(\boldsymbol{V}_2)$, will generate an equivalence class of points called a *coset*. The union of all cosets is $\mathrm{LAT}(\boldsymbol{V}_1)$.

Thus the concept of a coset will provide a natural way to partition $\mathrm{LAT}(\boldsymbol{V}_1)$ into subsets, which is a necessary step for the generation of multidimensional multirate filter banks.

Let $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ be integer-valued matrices such that

$$\mathrm{LAT}(\boldsymbol{V}_2) \subseteq \mathrm{LAT}(\boldsymbol{V}_1)$$

Let $\boldsymbol{a} \in \mathrm{LAT}(\boldsymbol{V}_1) \cap \mathrm{FPD}(\boldsymbol{V}_2)$. Define the coset $C(\boldsymbol{V}_1, \boldsymbol{V}_2, \boldsymbol{a})$ to be

$$C(\boldsymbol{V}_1, \boldsymbol{V}_2, \boldsymbol{a}) = \mathrm{LAT}(\boldsymbol{V}_2) + \boldsymbol{a}$$

If $\mathrm{LAT}(\boldsymbol{V}_1) = \mathcal{N}$, then by convention, $\boldsymbol{V}_1$ is not explicitly identified; that is, the coset is simply

$$C(\boldsymbol{V}_2, \boldsymbol{a}) = \mathrm{LAT}(\boldsymbol{V}_2) + \boldsymbol{a}$$

Given integer-valued matrices $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ such that $\mathrm{LAT}(\boldsymbol{V}_2) \subseteq \mathrm{LAT}(\boldsymbol{V}_1)$, we denote the set of shift vectors by

$$\mathcal{N}(\boldsymbol{V}_1, \boldsymbol{V}_2) = \{\boldsymbol{a} | \boldsymbol{a} \in \mathrm{LAT}(\boldsymbol{V}_1) \cap \mathrm{FPD}(\boldsymbol{V}_2)\}$$

Similar to the convention for cosets, the convention for shift vectors when $\mathrm{LAT}(\boldsymbol{V}_1) = \mathcal{N}$ is not to explicitly identify $\boldsymbol{V}_1$; that is, the set of shift vectors is simply

$$\mathcal{N}(\boldsymbol{V}_2) = \{\boldsymbol{a} | \boldsymbol{a} \in \mathcal{N} \cap \mathrm{FPD}(\boldsymbol{V}_2)\}$$

Returning to the example above, if

$$\boldsymbol{V}_2 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$

and

$$\boldsymbol{V}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

then the cosets are uniquely defined by the following set of shift vectors

$$\mathcal{N}(\boldsymbol{V}_1, \boldsymbol{V}_2) = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right\}$$

MULTIDIMENSIONAL SAMPLED SIGNALS

Unfortunately, some important sampling structures cannot be represented as a lattice. For example, consider an important sampling structure for high definition television (HDTV) called *line quincunx,* where two samples are placed one verti-

cally above the other in place of every sample in the sampling grid. But line quincunx can be represented as the union of two shifted lattices using the multidimensional z-transform. Here we first define some underlying vector mathematics, and then we present the multidimensional z-transform and the multidimensional discrete Fourier transform (DFT). In order to generalize the definitions that we have grown accustomed to seeing in one-dimension, we provide the definition of a vector raised to a vector power and subsequently the definition of a vector raised to a matrix power.

Given complex-valued vector

$$\boldsymbol{r} = \left[r_0, \ldots, r_{N-1}\right]^{\mathsf{T}}$$

and integer-valued vector

$$\boldsymbol{s} = \left[s_0, \ldots, s_{N-1}\right]^{\mathsf{T}}$$

Then the vector $\boldsymbol{r}$ raised to the vector $\boldsymbol{s}$ power is a scalar, and it is defined to be

$$\boldsymbol{r}^{\boldsymbol{s}} = r_0^{s_0}, r_1^{s_1}, \ldots, r_{N-1}^{s_{N-1}}$$

or equivalently,

$$\boldsymbol{r}^{\boldsymbol{s}} = \prod_{m=0}^{N-1} r_m^{\boldsymbol{s}_m}$$

Then, building on this definition, we can define a vector raised to a matrix power.

Given a complex-valued vector

$$\boldsymbol{r} = [r_0, \ldots, r_{N-1}]^{\mathsf{T}}$$

and an integer-valued matrix

$$\boldsymbol{L} = [L_0, \ldots, L_{N-1}]$$

where L_i is the ith column of $\boldsymbol{L}$. Then the vector $\boldsymbol{r}$ raised to the matrix $\boldsymbol{L}$ power is a row vector, and it is defined to be

$$\boldsymbol{r}^{\boldsymbol{L}} = [\boldsymbol{r}^{\boldsymbol{L}_0}, \boldsymbol{r}^{\boldsymbol{L}_1}, \ldots, \boldsymbol{r}^{\boldsymbol{L}_{N-1}}]$$

Multidimensional z-Transform

The k-dimensional z-transform of $x(n_0, \ldots, n_{k-1})$ is defined by

$$X(z_0, \ldots, z_{k-1}) = \sum_{n_0} \cdots \sum_{n_{k-1}} x(n_0, \ldots, n_{k-1}) z_0^{-n_0} \ldots z_{k-1}^{-n_{k-1}}$$

or equivalently,

$$X(\boldsymbol{z}) = \sum_{\boldsymbol{n} \in \mathcal{N}} \mathrm{x}(\boldsymbol{n}) \boldsymbol{z}^{-n}$$

where, $\boldsymbol{z} = [\boldsymbol{z}_0, \ldots, \boldsymbol{z}_{k-1}]^{\mathsf{T}}$ is a complex-valued vector, $\boldsymbol{n} = [\boldsymbol{n}_0, \ldots, \boldsymbol{n}_{k-1}]^{\mathsf{T}}$ is an integer-valued vector, and

$$\boldsymbol{z}^{-\boldsymbol{n}} = \prod_{m=0}^{k-1} z_m^{-\boldsymbol{n}_m}$$

Let $\boldsymbol{L}$ be an integer-valued nonsingular matrix, then, $\boldsymbol{z}^{\boldsymbol{L}}$ is given by

$$\boldsymbol{z}^{\boldsymbol{L}} = [\boldsymbol{z}^{\boldsymbol{L}_0}, \ldots, \boldsymbol{z}^{\boldsymbol{L}_{k-1}}]$$

where L_i is the ith column of $\boldsymbol{L}$, that is,

$$\boldsymbol{L} = [\boldsymbol{L}_0, \ldots, \boldsymbol{L}_{k-1}]$$

The following interesting property of the multidimensional z-transform can be easily shown. Let $\boldsymbol{L}$ be an integer-valued matrix where L_i, $i = 0, \ldots, k-1$, are the columns of $\boldsymbol{L}$. Then

$$(\boldsymbol{z}^{\boldsymbol{L}})^{\boldsymbol{n}} = \boldsymbol{z}^{\boldsymbol{L}\boldsymbol{n}}$$

Multidimensional Discrete Fourier Transform

The multidimensional discrete Fourier transform (DFT) is an exact Fourier representation for periodically sampled arrays. Therefore, it takes the form of a periodically sampled Fourier transform. As in the one-dimensional case, the multidimensional discrete Fourier transform can be interpreted as a Fourier series representation for one period of a periodic sequence.

In this formulation we have to address two types of periodicities: one due to the sampling lattice and one due to the signal (defined on lattice points) to be Fourier transformed. Let $\boldsymbol{V}$ denote the sampling matrix; hexagonal, quincunx, rectangular, and so on. Let $\boldsymbol{N}$ denote the periodicity matrix, which characterizes the periodicity of the lattice points on which the signal to be Fourier transformed is defined. Assume that LAT($\boldsymbol{N}$) is a sublattice of LAT($\boldsymbol{V}$). Then we define equivalence classes between periodic replicas of the data by

$$[\boldsymbol{n}] = \{\boldsymbol{m} \in \mathrm{LAT}(\boldsymbol{V}) | \boldsymbol{n} - \boldsymbol{m} \in \mathrm{LAT}(\boldsymbol{N})\}$$

Therefore, if parallelograms are drawn between the elements of LAT($\boldsymbol{N}$), then any two vectors that occupy the same relative position are in the same equivalence class.

Many properties of the periodicity matrix, $\boldsymbol{N}$, follow by analogy from the corresponding facts for sampling matrices. For example, the density of the periodicity matrix is uniquely defined by $1/|\det \boldsymbol{N}|$, denoted $1/\mathrm{J}(\boldsymbol{N})$, but for a given periodic sequence the periodicity matrix $\boldsymbol{N}$ is not unique, since it can be multiplied by any unimodular matrix and still describe the same periodic signal. In addition the columns of $\boldsymbol{N}$ indicate the vectors along which it is periodically replicated.

A multidimensional sequence $x(\boldsymbol{n})$ is periodic with period $\boldsymbol{N}$; that is, for all $\boldsymbol{n}, \boldsymbol{r} \in \mathcal{N}$, $x(\boldsymbol{n}) = \boldsymbol{x}(\boldsymbol{n} + \boldsymbol{N}\boldsymbol{r})$. Let $\mathcal{I}_N$ represent one period of $x(\boldsymbol{n})$. Then

$$X(\underline{\omega}) = \sum_{\boldsymbol{n} \in \mathcal{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V} \boldsymbol{n}]$$

where $\boldsymbol{V}$ defines the underlying sampling lattice. Moreover, since $x(\boldsymbol{n})$ is periodic with period $\boldsymbol{N}$, $X(\underline{\omega})$ can also be written as

$$X(\underline{\omega}) = \sum_{\boldsymbol{n} \in \mathcal{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V} (\boldsymbol{n} + \boldsymbol{N}\boldsymbol{r})]$$

or simply as

$$X(\underline{\omega}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V}\boldsymbol{n}] \exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V}\boldsymbol{N}\boldsymbol{r}]$$

But

$$X(\underline{\omega}) = \sum_{\boldsymbol{n}_i \in \mathscr{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V}\boldsymbol{n}]$$

Therefore

$$\exp[-j\underline{\omega}^{\mathsf{T}} \boldsymbol{V}\boldsymbol{N}\boldsymbol{r}] = 1$$

which is equivalent to the condition

$$\underline{\omega}^{\mathsf{T}} \boldsymbol{V}\boldsymbol{N} = 2\pi \mathbf{m}^{\mathsf{T}}$$

where $\boldsymbol{m}$ is a vector of integers. Upon further examination of $\underline{\omega}^{\mathsf{T}}$, we observe that

$$\underline{\omega}^{\mathsf{T}} = 2\pi \boldsymbol{m}^{\mathsf{T}} (\boldsymbol{V}\boldsymbol{N})^{-1}$$

or equivalently,

$$\underline{\omega}^{\mathsf{T}} = (2\pi (\boldsymbol{V}\boldsymbol{N})^{-\mathsf{T}} \boldsymbol{m})^{\mathsf{T}}$$

Therefore

$$\underline{\omega} = 2\pi (\boldsymbol{V}\boldsymbol{N})^{-\mathsf{T}} \boldsymbol{m}$$

The matrix $2\pi(\boldsymbol{V}\boldsymbol{N})^{-\mathsf{T}}$ serves as a Fourier domain sampling matrix. Substituting this equation into the equation for $X(\underline{\omega})$ yields

$$X(\boldsymbol{m}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j2\pi \boldsymbol{m}^{\mathsf{T}} \boldsymbol{N}^{-1} \boldsymbol{V}^{-1} \boldsymbol{V}\boldsymbol{n}]$$

or equivalently,

$$X(\boldsymbol{m}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-jm^{\mathsf{T}} (2\pi \boldsymbol{N}^{-1}) \boldsymbol{n}]$$

Let us further examine the inner product that occurs in the argument of the exponential:

$$\begin{aligned} \boldsymbol{m}^{\mathsf{T}} (2\pi \boldsymbol{N})^{-1} \boldsymbol{n} &= (2\pi \boldsymbol{N}^{-1} \boldsymbol{n})^{\mathsf{T}} \boldsymbol{m} \\ &= \boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m} \end{aligned}$$

Therefore

$$X(\boldsymbol{m}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\mathrm{N}}} x(\boldsymbol{n}) \exp[-j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}]$$

Suppose that the multidimensional sequence $X(\boldsymbol{m})$ is periodic with period $\boldsymbol{P}$; that is, $X(\boldsymbol{m}) = \boldsymbol{X}(\boldsymbol{m} + \boldsymbol{P}\boldsymbol{q})$ for $\boldsymbol{m}, \boldsymbol{q} \in \mathscr{N}$. Also let $\mathscr{I}_{\mathbf{P}}$ represent one period of $X(\boldsymbol{m})$. Then, by analogy with the one-dimensional discrete Fourier transform, $x(\boldsymbol{n})$ has the following form for some constant α:

$$x(\boldsymbol{n}) = \frac{1}{\alpha} \sum_{\mathrm{m} \in \mathscr{I}_{\mathrm{P}}} \boldsymbol{X}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) m]$$

Invoking the periodicity of $X(\boldsymbol{m})$, that is, $X(\boldsymbol{m}) = \boldsymbol{X}(\boldsymbol{m} + \boldsymbol{P}\boldsymbol{q})$, will cause $x(\boldsymbol{n})$ to become

$$x(\boldsymbol{n}) = \frac{1}{\alpha} \sum_{\boldsymbol{m} \in \mathscr{I}_{\mathrm{P}}} \boldsymbol{X}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) (\boldsymbol{m} + \boldsymbol{P}\boldsymbol{q})]$$

or equivalently,

$$x(\boldsymbol{n}) = \frac{1}{\alpha} \sum_{\boldsymbol{m} \in \mathscr{I}_{\boldsymbol{P}}} \boldsymbol{X}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}] \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{P}\boldsymbol{q}]$$

But

$$x(\boldsymbol{n}) = \frac{1}{\alpha} \sum_{\boldsymbol{m} \in \mathscr{I}_{\boldsymbol{P}}} \boldsymbol{X}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}]$$

Therefore

$$\exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{P}\boldsymbol{q}] = 1 \qquad \text{for all } \boldsymbol{q} \in \mathscr{N}$$

Since $\boldsymbol{n}$ and $\boldsymbol{q}$ are integer-valued vectors, then

$$\boldsymbol{N}^{-\mathsf{T}} \boldsymbol{P} = \boldsymbol{I}$$

or equivalently,

$$\boldsymbol{P} = \boldsymbol{N}^{\mathsf{T}}$$

Therefore, $X(\boldsymbol{m})$ is periodic with period $\boldsymbol{N}^{\mathsf{T}}$; that is, $X(\boldsymbol{m}) = \boldsymbol{X}(\boldsymbol{m} + \boldsymbol{N}^{\mathsf{T}}\boldsymbol{q})$. Hence

$$x(\boldsymbol{n}) = \frac{1}{\alpha} \sum_{\boldsymbol{m} \in \mathscr{I}_{\boldsymbol{N}^{\mathsf{T}}}} \boldsymbol{X}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}]$$

Now let us determine the constant α by substituting the equation for $x(\boldsymbol{n})$ into the equation for $X(\boldsymbol{m})$. Hence

$$X(\boldsymbol{m}) = \frac{1}{\alpha} \sum_{\boldsymbol{s} \in \mathscr{I}_{\boldsymbol{N}^{\mathsf{T}}}} \boldsymbol{X}(\boldsymbol{s}) \sum_{\boldsymbol{n} \in \mathscr{I}_{\boldsymbol{N}}} \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}] \exp[-j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{s}]$$

or equivalently,

$$X(\boldsymbol{m}) = \frac{1}{\alpha} \sum_{\boldsymbol{s} \in \mathscr{I}_{\boldsymbol{N}^{\mathsf{T}}}} \boldsymbol{X}(\boldsymbol{s}) \sum_{\boldsymbol{n} \in \mathscr{I}_{\boldsymbol{N}}} \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) (\boldsymbol{m} - \boldsymbol{s})]$$

However,

$$\sum_{\boldsymbol{n} \in \mathscr{I}_{\boldsymbol{N}}} \exp[\, j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) (\boldsymbol{m} - \boldsymbol{s})] = J(\boldsymbol{N}) \delta_{\boldsymbol{m}, \boldsymbol{s}}$$

Hence

$$\alpha = J(\boldsymbol{N})$$

which is as expected, since $\mathrm{J}(\boldsymbol{N}) = |\det \boldsymbol{N}|$ is the number of samples in one period for LAT($\boldsymbol{N}$). Therefore the multidimensional discrete Fourier transform pair are given by

$$X(\boldsymbol{m}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\boldsymbol{N}}} \boldsymbol{x}(\boldsymbol{n}) \exp[-j\boldsymbol{n}^{\mathsf{T}} (2\pi \boldsymbol{N}^{-\mathsf{T}}) \boldsymbol{m}]$$

and

$$x(\boldsymbol{n}) = \frac{1}{J(\boldsymbol{N})} \sum_{\boldsymbol{m} \in \mathscr{I}_{\boldsymbol{N}^{\mathsf{T}}}} \boldsymbol{N}(\boldsymbol{m}) \exp[\, j\boldsymbol{n}^{\mathsf{T}}(2\pi \boldsymbol{N}^{-\mathsf{T}})\boldsymbol{m}]$$

It should be noted that these equations reduce to the usual discrete Fourier transform pair in the one-dimensional case and to the familiar rectangular multidimensional discrete Fourier transform when $\boldsymbol{N}$ is a diagonal matrix.

As an illustration of this theoretical development, sometimes it is of interest to input data from an arbitrary lattice and output it on a rectangular lattice so that it could be conveniently displayed on a computer display. Assume that $\boldsymbol{V}$ is defined by

$$\boldsymbol{V} = \begin{bmatrix} a & b \\ 0 & c \end{bmatrix}$$

For hexagonal input, $a = 2$, $b = 1$, and $c = 2$. Moreover, for quincunx input, $a = 2$, $b = 1$, and $c = 1$. In addition, for rectangular input, $b = 0$. Select a periodicity matrix so that $\boldsymbol{VN}$ is a diagonal matrix. The resulting Fourier analysis will be on a rectangular grid. Now let us pick $\boldsymbol{N}$ to be

$$\boldsymbol{N} = \begin{bmatrix} N_1 & -\dfrac{b}{a}N_2 \\ 0 & N_2 \end{bmatrix}$$

Then

$$\boldsymbol{VN} = \begin{bmatrix} a & b \\ 0 & c \end{bmatrix} \begin{bmatrix} N_1 & -\dfrac{b}{a}N_2 \\ 0 & N_2 \end{bmatrix} = \begin{bmatrix} aN_1 & 0 \\ 0 & cN_2 \end{bmatrix}$$

This $\boldsymbol{N}$ matrix is a good choice for a periodicity matrix. Therefore the DFT becomes

$$X(\boldsymbol{m}) = \sum_{\boldsymbol{n} \in \mathscr{I}_{\boldsymbol{N}}} \boldsymbol{x}(\boldsymbol{n}) \exp[-\boldsymbol{j}\boldsymbol{n}^{\mathsf{T}}(2\pi \boldsymbol{N}^{-\mathsf{T}})\boldsymbol{m}]$$

where

$$\boldsymbol{N}^{-\mathsf{T}} = \begin{bmatrix} \dfrac{1}{N_1} & 0 \\ \dfrac{b}{aN_1} & \dfrac{1}{N_2} \end{bmatrix}$$

This suggests the following algorithm:

1. Compute N_2N_1-point fast Fourier transforms (FFTs), one for each row in the n_1 direction:

$$X_1(m_1, n_2) = \sum_{n_1=0}^{N_1-1} x(n_1, n_2) \exp\left(-j2\pi \frac{n_1 m_1}{N_1}\right)$$

2. Apply a phase shift to each point of the resulting data:

$$X_2(m_1, n_2) = X_1(m_1, n_2) \exp\left(-j\frac{2\pi}{N_1} \operatorname{round}\left(\frac{b}{a} n_2 m_1\right)\right)$$

Since we are working with a sampling grid with samples at integer-valued locations, it is important that we perform the phase shift for integer multiples of $2\pi/N_1$. But b/an_2m_1 is real-valued. Therefore we need to quantize b/an_2m_1 to integer values through the use of the round function.

3. Compute N_1N_2-point FFTs, one for each column in the n_2 direction:

$$X_3(m_1, m_2) = \sum_{n_2=0}^{N_2-1} X_2(m_1, n_2) \exp\left(-j2\pi \frac{n_2 m_2}{N_2}\right)$$

The last example has shown how one-dimensional techniques can be generalized to perform two-dimensional operations. This foreshadows the work in the following section on system identification. Much of this work has been and continues to be done with one-dimensional problems, but as more sophisticated problems emerge, the underlying geometry will need to be considered. This will lead the area of system identification into multidimensional statistical signal processing.

SYSTEM IDENTIFICATION PROBLEM

Determining a *dynamical* system's parameters from its *noise*-corrupted input and output measurements is what system identification is all about. As such, system identification stands out in stark contrast to the mathematical modeling-based approaches to dynamical system elucidation, so engrained in physics and engineering practice, for system identification, embraces an empiricism-based route to modeling. Therefore system identification is a basic scientific tool, for it entails a "black box" approach to modeling. In other words, a model of the dynamical system is being matched to the known input data and the measured output of the system. In the development here, it is envisaged that the data (input and output) are specified on a simple one-dimensional and regular grid, yet from the earlier discussion it should become apparent how the system identification methodology presented in the sequel can be adapted to the multidimensional setting.

Linear discrete-time single input/single output (SISO) control systems are considered, and their transfer function

$$\frac{y(z)}{u(z)} = \frac{b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m}}{1 - a_1 z^{-1} - \cdots - a_n z^{-n}}$$

is identified; that is, the $n + m$ coefficients $a_1, \ldots, a_n, b_1, \ldots, b_m$ are determined. The corruption of the input (u) and the output (y) by measurement noise is a major concern, and therefore system identification entails a statistical approach to modeling. Hence it should come as no surprise that the methods of statistics have a strong bearing on system identification, as shown in this article. Roughly speaking, system identification is the "dynamic" counterpart of the "static" linear regression method of statistics, whose broad fields of application encompass the "softer" (i.e., with less structure) endowed disciplines of economics and the social sciences. Hence, because of its statistical foundations, system identification is applicable to a wide variety of economic, scientific, and engineering problems. However, if system identification were a straightforward task, our dependence on mathematical modeling, and indeed on physics, would be significantly reduced.

Unfortunately, the inverse problem nature of system identification requires that careful attention be given to it. So far regression-based system identification has not been widely applied [e.g., see (7,8)]. In this article the current shortcomings of the system identification paradigm are elucidated, and a practical system identification algorithm is developed. Validation issues are also addressed.

Linear regression based approaches for the identification of the parameters of linear control systems are used in on-line and real-time operations where the linear structure of the dynamics is directly exploited and only the system's parameter (without the system's state) is estimated. In the signal processing literature, this main line of research in system identification based on the statistical method of linear regression (10) is also referred to as "linear prediction."

If either the system under consideration is static, as is the case in the linear regression paradigm of statistics, or if dynamical systems with process noise but with no measurement noise are considered, then auto regressive with exogeneous inputs (ARX) models are obtained. The problem of estimating the parameters of an ARX model leads to a linear regression formulation, whose solution is given by a least-squares estimate. Therefore the identification of the parameters of an ARX model is a relatively simple task. These models are often discussed in the controls literature. Unfortunately, ARX models are not very interesting in control work, due to the dynamic nature of control systems and the ubiquity of measurement noise. Note that in signal processing, finite impulse response (FIR) filters yield ARX models.

Linear regression–based system identification algorithms applied to the identification of discrete-time *dynamical* systems with *measurement* noise yield auto regressive moving average with exogeneous inputs (ARMAX) models.

It is important to recognize that, notwithstanding the linear structure of the linear regression, the identification of an ARMAX model is a nonlinear filtering problem. The linear regression, like the formulation of equations to be solved in order to identify (determine) the parameter vector, only serves to mask the inherently nonlinear nature of the original system identification problem. Thus in ARMAX models measurement (or sensor) noise is responsible for the introduction of correlation into the "equation error" of the ensuing "linear" regression. Correlation causes the least squares–based parameter estimates to be "biased," which means that the parameter estimates are bad. Hence, when ARMAX models are used, it is important to recognize and properly model the correlation inherent in the linear regression's equation error. It is therefore required to calculate the parameter's minimum variance estimate, which incorporates the equation error covariance information. In conclusion, the notorious correlation phenomenon encountered in the "linear" regression formulation of the problem of identifying ARMAX models is just an alternative manifestation of the difficult nature of the nonlinear filtering problem.

These basic difficulties of system identification point to the following two avenues of approach, which have been explored with varied success. Roughly speaking, the trade-off is between computational effort and instrumentation hardware:

1. *Simple Static Estimation.* The identification problem of dynamic systems in the presence of sensor noise is transformed into a static estimation problem provided that additional variables are being measured. This approach is feasible provided that additional sensors are used. This affords the use of ARX models for the identification of dynamic systems with sensor noise. The inclusion of additional sensors reduces the computational effort. This approach is therefore particularly suitable for on-line system identification, as required in adaptive and reconfigurable control. This approach is successfully pursued in Refs. 11–14.
2. *ARMAX Models with Dynamic System Identification.* A careful analysis of the attendant stochastic problem is required, involving discrete measurements and developing iterative and computationally intensive algorithms.

In this article the second approach is emphasized.

Linear Regression

Static estimation problems are the object of statistics and are referred to as linear regression problems. Consider the static linear regression problem where the parameter vector $\Theta \in R^n$ needs to be estimated

$$Z = H\Theta + V \tag{1}$$

The "measurement vector" is $Z \in R^N$, and the known regressor H is an $N \times n$ matrix. The statistics of the "equation error," or measurement noise, $V \in R^N$ are specified: V is a zero-mean Gaussian random vector whose known covariance matrix

$$R = E(VV^{\mathsf{T}}) \tag{2}$$

R is an $N \times N$ real, symmetric and positive definite matrix.

The minimum variance (MV) parameter estimate is

$$\hat{\Theta}_{\mathrm{MV}} = (H^{\mathsf{T}}R^{-1}H)^{-1}H^{\mathsf{T}}R^{-1}Z \tag{3}$$

The estimation error covariance

$$E((\Theta - \hat{\Theta})(\Theta - \hat{\Theta})^{\mathsf{T}}) = P_{\mathrm{MV}}$$

where the $n \times n$ real, symmetric, and positive definite matrix

$$P_{\mathrm{MV}} = (H^{\mathsf{T}}R^{-1}H)^{-1} \tag{4}$$

The following is an important special case: The covariance matrix of the equation error is a scaled identity matrix

$$R = rI_N$$

where r is a positive number. In this case the parameter estimate is particularly simple:

$$\hat{\Theta}_{\mathrm{LS}} = (H^{\mathsf{T}}H)^{-1}H^{\mathsf{T}}Z \tag{5}$$

and the estimation error covariance is

$$P_{\mathrm{LS}} = r(H^{\mathsf{T}}H)^{-1} \tag{6}$$

The estimate Eq. (5) is also referred to as the least-squares (LS) estimate.

The crucial advantage of the LS parameter estimate Eq. (5) is its independence from the covariance of the equation error, which in this case is determined solely by the measurement noise intensity r. Furthermore, note that in this important special case where the measurement's error covariance matrix is a scaled identity matrix, the LS estimate *is* in fact the MV estimate.

The system identification route to parameter estimation is rooted in the statistical method of linear regression (10). Linear regression is basically a batch-type algorithm. Hence, the system identification algorithms developed in the sequel are readily adaptable to a "moving window" type of algorithm, and hence they are used to estimate time-varying parameters and parameters subject to jumps. A useful rule of thumb from statistics is to take a large number of measurements for the batch size:

$$N \approx n^2$$

Thus the batch data processing to system identification is the preferred approach.

Linear Regression for System Identification 1

An nth order linear discrete-time SISO control system is considered. The dynamical system is

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + \cdots + a_n y_{k-n+1} + b_1 u_k + b_2 u_{k-1} + \cdots + b_m u_{k-m+1},\ k = 1, 2, \ldots \tag{7}$$

The measurement is

$$z_{k+1} = y_{k+1} + v_{k+1} \tag{8}$$

where the measurement noise v_{k+1} is a Gaussian random variable with a variance of σ^2. The measurement noise is white, $E(v_k v_l) = 0$ for all $k \neq l$ (i.e., there is no correlation).

A naive linear regression approach to system identification entails the "substitution" of Eq. (8) into Eq. (7) so that

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + \cdots + a_n y_{k-n+1} + b_1 u_k + \cdots + b_m u_{k-m+1} + v_{k+1} \tag{9}$$

is obtained. This is indeed an ARX model.

Concatenating N measurements yields the linear regression

$$\begin{bmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N} \end{bmatrix} = \begin{bmatrix} y_k & y_{k-1} & \cdots & y_{k-n+1} \\ y_{k+1} & y_k & \cdots & y_{k-n+2} \\ \vdots & \vdots & & \vdots \\ y_{k+N-1} & y_{k+N-2} & \cdots & y_{k+N-n} \end{bmatrix} \begin{bmatrix} u_k & u_{k-1} & \cdots & u_{k-m+1} \\ u_{k+1} & y_k & \cdots & u_{k-m+2} \\ \vdots & \vdots & & \vdots \\ u_{k+N-1} & u_{k+N-2} & \cdots & u_{k-m+N-1} \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_n \end{bmatrix} + \begin{bmatrix} v_{k+1} \\ v_{k+2} \\ \vdots \\ v_{k+N-1} \end{bmatrix} \tag{10}$$

Next, define

$$Z = \begin{bmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N} \end{bmatrix}_{N\times 1} \tag{11}$$

$$V = \begin{bmatrix} v_{k+1} \\ v_{k+2} \\ \vdots \\ v_{k+N} \end{bmatrix}_{N\times 1} \tag{12}$$

and

$$H = \begin{bmatrix} y_k & y_{k-1} & \cdots & y_{k-n+1} \\ y_{k+1} & y_k & \cdots & y_{k-n+2} \\ \vdots & \vdots & & \vdots \\ y_{k+N-1} & y_{k+N-2} & \cdots & y_{k+N-n} \end{bmatrix} \begin{bmatrix} u_k & u_{k-1} & \cdots & u_{k-m+1} \\ u_{k+1} & u_k & \cdots & u_{k-m+2} \\ \vdots & \vdots & & \vdots \\ u_{k+N-1} & u_{k+N-2} & \cdots & u_{k-m+N} \end{bmatrix}_{N\times(m+n)} \tag{13}$$

The parameter vector is

$$\Theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}_{(m+n)\times 1}$$

Thus the linear regression model Eq. (1) is obtained, where the covariance of the "measurement error" is a scaled unity matrix, namely

$$R = E(VV^{\mathrm{T}}) = \sigma^2 I_N$$

Hence the minimum variance estimate is the least-squares estimate given by Eq. (5). The estimation error covariance is given by Eq. (6).

In the conventional system identification literature, an inordinate amount of attention is being given to the recursive (on the number of recorded measurements N) form of the above result. One then refers to recursive least squares (RLS) system identification. The latter is readily derived using the Bayes formula. Thus, given the parameter estimate $\hat{\Theta}_N$ and the estimation error covariance matrix P_N, which are obtained after a data record of length N has been processed, the latest $N + 1$ measurement satisfies the scalar equation

$$y_{k+N+1} = h\Theta + v_{k+N+1} \tag{14}$$

where the row vector

$$h_{1\times(m+n)} = (y_{k+N}, y_{k+N-1}, \ldots, y_{k+N-n+1}, u_{k+N}, u_{k+N_1}, \ldots, u_{k-m+N}) \tag{15}$$

The $(N+1)$th measurement is integrated into the estimation algorithm as follows:

$$\hat{\Theta}_{N+1} = \hat{\Theta}_N + K(y_{k+N+1} - h\hat{\Theta}_N) \tag{16}$$

where the Kalman gain

$$K = \frac{1}{hP_Nh^T + \sigma^2} P_N h^{\mathrm{T}} \tag{17}$$

and the covariance of the updated estimation error is

$$P_{N+1} = P_N - \frac{1}{hP_Nh^T + \sigma^2} P_N h^{\mathrm{T}} h P_N \tag{18}$$

Note that additional measurements help improve the parameter estimate, so $P_{N+1} \leq P_N$, as expected. At the same time the parameter estimate supplied by the RLS algorithm at time N is identical to the parameter estimate arrived at by applying the batch linear regression algorithm to the very same data record (of length N) provided that the recursive algorithm was initialized at some earlier time $N' < N$ using the estimate and the estimation error covariance supplied by an application of the batch algorithm to an initial data record of length N'. This result follows from the application of the matrix inversion lemma (15).

The RLS- and LS-based system identification algorithms are widely used in the control community to identify the parameters of control systems [specified by Eqs. (7) and (8)]. At the same time their estimation performance is often deficient. For this reason the parameter estimate is euphemistically referred to as biased, so the RLS- or LS-based system identification does not work. The root cause of the failure of the RLS or the LS identification algorithms Eqs. (16)–(18) or (5) and (6), respectively, is the sloppy derivation of the linear regression in Eq. (9). Hence in a later section, in sequel, a proper analysis is undertaken.

Linear Regression for System Identification 2

For the ARMAX model a careful stochastic analysis of the parameter estimation process is required. In this respect, the distinction between the true output of the control system at time k, y_k, and the actually recorded measurement, z_k, is crucial. Thus, y_k is an internal variable governed by the dynamics Eq. (7) and is not directly accessible to the observer. The observer records the measurements z_k which are related to the internal variable y_k according to the measurement Eq. (8). It is here assumed that the input variable u_k is noiseless.

Use the measurement Eq. (8) to back out the internal variable y_l, for $l = k - n + 1, \ldots, k + N$ [e.g., see Eq. (7)]. Thus

$$\begin{aligned} y_{k-n+1} &= z_{k-n+1} - v_{k-n+1} \\ &\vdots \\ y_{k+1} &= z_{k+1} - v_{k+1} \\ y_k &= z_k - v_k \\ &\vdots \\ y_{k+N} &= z_{k+N} - v_{k+N} \end{aligned} \tag{19}$$

Next insert Eq. (19) into Eq. (7). Define the zero mean Gaussian random variable

$$\tilde{v}_l = v_l - a_1 v_{l-1} - a_2 v_{l-2} - \cdots - a_n v_{l-n}, \qquad l = k+1, \ldots, k+N \tag{20}$$

Hence the novel linear regression is obtained

$$z_{l+1} = a_1 z_l + a_2 z_{l-1} + \cdots + a_n z_{l-n+1} + b_1 u_l + b_2 u_{l-1} + \cdots + b_m u_{l-m+1} + \tilde{v}_{l+1}, \qquad l = k, \ldots, k+N-1 \tag{21}$$

The linear regression Eq. (21) is in appearance similar to the linear regression Eq. (9). However, the entries of the Z and H matrices now consist of the actual measurements/observables z, and not the unavailable internal variables y:

$$Z = \begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix}_{N\times 1}$$

$$H = \begin{bmatrix} z_k & z_{k-1} & \cdots & z_{k-n+1} \\ z_{k+1} & z_k & \cdots & z_{k-n+2} \\ \vdots & \vdots & & \vdots \\ z_{k+N-1} & z_{k+N-2} & \cdots & z_{k+N-n} \end{bmatrix} \begin{bmatrix} u_k & u_{k-1} & \cdots & u_{k-m+1} \\ u_{k+1} & u_k & \cdots & u_{k-m+2} \\ \vdots & \vdots & & \vdots \\ u_{k+N-1} & u_{k+N-2} & \cdots & u_{k-m+N} \end{bmatrix}_{N\times(n+m+2)} \tag{22}$$

Moreover, the "equation error" in Eq. (21) is the zero-mean Gaussian random variable

$$\tilde{V} = \begin{bmatrix} \tilde{v}_{l+1} \\ \vdots \\ \tilde{v}_{l+N} \end{bmatrix} \tag{23}$$

Now the calculation of the minimum variance estimate of the parameter associated with the linear regression Eq. (21) requires the evaluation of the covariance of the equation error. Hence the expectation

$$R = E(\tilde{V}\tilde{V}^{\mathrm{T}}) \tag{24}$$

needs to be calculated. The elements of the real symmetric and positive (semi)definite R matrix are calculated by invoking Eq. (20). Thus, the diagonal elements of the R matrix are all equal:

$$\begin{aligned} R_{i,i} &= E(\tilde{v}_{k+i}^2) = E((v_{k+i} - a_1 v_{k+i-1} - \cdots - a_n v_{k+i-n})^2) \\ &= r = \sigma^2(1 + \sum_{k=1}^{n} a_k^2) \end{aligned} \tag{25}$$

for all $i = 1, \ldots, N$. The off-diagonal elements of the symmetric equation error covariance matrix are

$$R_{i,j} = E(\tilde{v}_{k+i}\tilde{v}_{k+j}) = \sigma^2 \left(-a_{i-j} + \sum_{k=1}^{n+j-i} a_k a_{i-j+k} \right) \tag{26}$$

for all $i = 1, \ldots, N, j = 1, \ldots, N$ and $i > j$. For example, the 1, 2 element of the equation error covariance matrix is

$$\begin{aligned} R_{1,2} &= R_{2,1} = E(\tilde{v}_{k+1}\tilde{v}_{k+2}) \\ &= E((v_{k+1} - a_1 v_k - a_2 v_{k-1} - \cdots - a_n v_{k-n+1}) \\ &\quad (v_{k+2} - a_1 v_{k+1} - a_2 v_k - \cdots - a_n v_{k-n+2})) \\ &= \sigma^2(-a_1 + a_1 a_2 + a_2 a_3 + \cdots + a_{n-1} a_n) \end{aligned}$$

The off-diagonal elements of R no longer vanish. In other words, the equation error random vector $\tilde{V}$ is not white, for example, $E(\tilde{V}_1 \tilde{V}_2) \neq 0$, and there is correlation in $\tilde{V}$. Correlation is responsible for the fact that the least-squares and minimum-variance estimates are no longer identical, since in Eq. (3) the R matrix is not a scaled identity matrix. Hence the least-squares formula Eq. (5) no longer yields the minimum variance estimate. That is why the widely used and easy-to-calculate least-squares estimate Eq. (5) is biased, that is, incorrect. The MV estimate Eq. (3) of Θ should be used instead.

The calculation of the minimum-variance estimate requires the knowledge of R. Unfortunately, R is not a priori known, and in addition to the expected dependence on the given sensor's measurement error σ, R is also determined by the (as yet unknown) coefficients of the system's transfer function denominator. Thus it is important to realize that [e.g., see Eq. (25) and (26)]

$$R = \sigma^2 R'(\Theta)$$

This calls for an iterative calculation of the minimum-variance estimate. Thus in Eqs. (25) and (26) the prior estimate $\hat{\Theta}_0$ of the parameter is used to estimate the covariance matrix R; then an improved minimum variance estimate of the parameter, Θ_1, is obtained from Eq. (3). Strictly speaking, only the prior estimates of the parameters of the system's dynamics, which are encapsulated in the coefficients $a_1, \ldots, a_n$, are used. Thus

$$R = \sigma^2 R' \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \tag{27}$$

This process is repeated, and the convergence of the so-obtained parameter sequence Θ_i is gauged. Numerical experimentation shows that when the above process converges then this parameter estimate closely approximates the true parameter.

SYSTEM IDENTIFICATION EXAMPLES

These concepts are carefully illustrated in the context of the identification of a first-order (scalar) control system.

Scalar Example

The dynamics are

$$y_{k+1} = a y_k + b u_k, \qquad k = 1, 2, \ldots \tag{28}$$

At time $k + 1$ the measurement equation is

$$z_{k+1} = y_{k+1} + v_{k+1} \tag{29}$$

The measurement noise v_{k+1} is a zero-mean Gaussian random variable with variance σ^2. The measurement errors v_k and v_l are temporally uncorrelated for all $k \neq l$.

The data record for time $k, k + 1, \ldots, k + N$ is considered, and following the analysis in the preceding section, the linear regression is obtained:

$$\begin{aligned} z_{k+1} &= a z_k + b u_k + \tilde{v}_{k+1} \\ z_{k+2} &= a z_{k+1} + b u_{k+1} + \tilde{v}_{k+2} \\ z_{k+N} &= a z_{k+N-1} + b u_{k+N-1} + \tilde{v}_{k+N} \end{aligned} \tag{30}$$

Let

$$Z = \begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix}, \quad H = \begin{bmatrix} z_k & u_k \\ z_{k+1} & u_{k+1} \\ \vdots & \vdots \\ z_{k+N-1} & u_{k+N-1} \end{bmatrix}, \quad \tilde{V} = \begin{bmatrix} \tilde{v}_{k+1} \\ \tilde{v}_{k+2} \\ \vdots \\ \tilde{v}_{k+N} \end{bmatrix}$$

and the parameter vector is

$$\Theta = \begin{bmatrix} a \\ b \end{bmatrix}$$

The covariance matrix of the "equation error" is

$$R = E(\tilde{V}\tilde{V}^{\mathrm{T}}) = E\left(\begin{bmatrix} \tilde{v}_{k+1} \\ \tilde{v}_{k+2} \\ \vdots \\ \tilde{v}_{k+N} \end{bmatrix} [\tilde{v}_{k+1} \quad \tilde{v}_{k+2} \quad \cdots \quad \tilde{v}_{k+N}] \right) \tag{31}$$

Hence

$$\begin{aligned} R = E\Bigg(& \begin{bmatrix} v_{k+1} - a v_k \\ v_{k+2} - a v_{k+1} \\ \vdots \\ v_{k+N} - a v_{k+N-1} \end{bmatrix} [v_{k+1} - a v_k \\ & \qquad v_{k+2} - a v_{k+1} \quad \cdots \quad v_{k+N} - a v_{k+N-1}] \Bigg) \end{aligned}$$

$$= \begin{bmatrix} E((v_{k+1} - a v_k)(v_{k+1} - a v_k)) & E((v_{k+1} - a v_k)(v_{k+2} - a v_k)) & \cdots \\ E((v_{k+2} - a v_{k+1})(v_{k+1} - a v_k)) & E((v_{k+2} - a v_{k+1})(v_{k+2} - a v_{k+1})) & \cdots \\ \vdots & & \\ E(v_{k+N} - a v_{k+N-1})(v_{k+1} - a v_k) & E(v_{k+N} - a v_{k+N-1})(v_{k+2} - a v_{k+1}) & \cdots \end{bmatrix} \tag{32}$$

Thus R is a tridiagonal $N \times N$ matrix:

$$R = \sigma^2 \begin{bmatrix} 1+a^2 & -a & 0 & 0 & 0 & 0 & 0 \\ -a & 1+a^2 & -a & 0 & 0 & 0 & 0 \\ 0 & -a & 1+a^2 & -a & 0 & 0 & 0 \\ 0 & 0 & & \cdots & & 0 & 0 \\ 0 & 0 & 0 & 0 & -a & 1+a^2 & -a \\ 0 & 0 & 0 & 0 & 0 & -a & 1+a^2 \end{bmatrix} \tag{33}$$

R is invertible, since for $a \neq 1$,

$$det(R) = \frac{a^{2(N+1)} - 1}{a^2 - 1}$$

and for $a = 1$,

$$det(R) = N + 1$$

Hence, in order to identify the parameters a and b of the ARMAX model above, the following iteration for the estimation of the control system's parameters is obtained:

$$\begin{bmatrix} \hat{a}_{i+1} \\ \hat{b}_{i+1} \end{bmatrix} = \left(\begin{bmatrix} z_k & z_{k+1} & \cdots & z_{k+N-1} \\ u_k & u_{k+1} & \cdots & u_{k+N-1} \end{bmatrix} \right.$$
$$\begin{bmatrix} 1+\hat{a}_i^2 & -\hat{a}_i & 0 & & & \\ -\hat{a}_i & 1+\hat{a}_i^2 & -\hat{a}_i & 0 & & \\ 0 & -\hat{a}_i & 1+\hat{a}_i^2 & -\hat{a}_i & 0 & \\ 0 & 0 & & \cdots & & \\ & & & 0 & -\hat{a}_i & 1+\hat{a}_i^2 \end{bmatrix}^{-1}$$
$$\left. \begin{bmatrix} z_k & u_k \\ z_{k+1} & u_{k+1} \\ \vdots & \vdots \\ z_{k+N-1} & u_{k+N-1} \end{bmatrix} \right)^{-1} \begin{bmatrix} z_k & z_{k+1} & \cdots & z_{k+N-1} \\ u_k & u_{k+1} & \cdots & u_{k+N_1} \end{bmatrix}$$
$$\begin{bmatrix} 1+\hat{a}_i^2 & -\hat{a}_i & 0 & 0 & 0 & 0 \\ -\hat{a}_i & 1+\hat{a}_i^2 & -\hat{a}_i & 0 & 0 & 0 \\ 0 & -\hat{a}_i & 1+\hat{a}_i^2 & -\hat{a}_i & 0 & 0 \\ & & & \cdots & & \\ 0 & 0 & 0 & 0 & 1+\hat{a}_i^2 & -\hat{a}_i \\ 0 & 0 & 0 & 0 & -\hat{a}_i & 1+\hat{a}_i^2 \end{bmatrix}^{-1}$$
$$\begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix} \tag{34}$$

The estimation error's covariance matrix is

$$P = \sigma^2 P' \tag{35}$$

where the 2×2 matrix P' is

$$\left(\begin{bmatrix} z_k & z_{k+1} & \cdots & z_{k+N-1} \\ u_k & u_{k+1} & \cdots & u_{k+N-1} \end{bmatrix} \right.$$
$$\begin{bmatrix} 1+\hat{a}^2 & -\hat{a} & 0 & 0 & 0 & 0 \\ -\hat{a} & 1+\hat{a}^2 & -\hat{a} & 0 & 0 & 0 \\ 0 & -\hat{a} & 1+\hat{a}^2 & -\hat{a} & 0 & 0 \\ 0 & 0 & & \cdots & & 0 \\ & & & 0 & -\hat{a} & 1+\hat{a}^2 \end{bmatrix}^{-1}$$
$$\left. \begin{bmatrix} z_k & u_k \\ z_{k+1} & u_{k+1} \\ \vdots & \vdots \\ z_{k+N-1} & u_{k+N-1} \end{bmatrix} \right) \tag{36}$$

For example, if two measurements are taken ($N = 2$), the following explicit parameter estimation formulas for a first-order ARMAX model are obtained:

$$P = \frac{\sigma^2}{(1+\hat{a}^2+\hat{a}^4)(u_k z_{k+1} - u_{k+1} z_k)} P'' \tag{37}$$

where the 2×2 P'' matrix is

$$\begin{bmatrix} -(1+\hat{a}^2)(u_k^2 + u_{k+1}^2) - 2\hat{a} u_k u_{k+1} \\ (1+\hat{a}^2)(u_k z_k + u_{k+1} z_{k+1}) + \hat{a}(u_k z_{k+1} + u_{k+1} z_k) \end{bmatrix}$$
$$\begin{bmatrix} (1+\hat{a}^2)(u_k z_k + u_{k+1} z_{k+1}) + \hat{a}(u_k z_{k+1} + u_{k+1} \\ -(1+\hat{a}^2)(z_k^2 + z_{k+1}^2) - 2\hat{a} z_k z_{k+1} \end{bmatrix}$$

and the parameter estimates are

$$\begin{aligned} \hat{a}_{MV_{i+1}} = {} & \frac{1}{(u_k z_{k+1} - u_{k+1} z_k)^2} [(1+\hat{a}_i^2 + \hat{a}_i^4)(u_k z_{k+1} - u_{k+1} z_k) \\ & (u_k z_{k+2} - u_{k+1} z_{k+1}) \\ & + 2\hat{a}_i (1+\hat{a}_i^2)(u_k z_{k+1} + u_{k+1} z_{k+2})(u_k z_{k+1} + u_{k+1} z_k)] \\ \hat{b}_{MV_{i+1}} = {} & \frac{1}{(u_k z_{k+1} - u_{k+1} z_k)^2} [(1+a^2)(z_k^2 + z_{k+1}^2) + 2a z_k z_{k+1}] \\ & \times [(1+a^2)(u_k z_{k+1} + u_{k+1} z_{k+2}) + a(u_k z_{k+2} + u_{k+1} z_{k+1})] \\ & - [(1+a^2)(u_k z_k + u_{k+1} z_{k+1}) - a(u_k z_{k+1} + z_k u_{k+1})] \\ & \times [(1+a^2)(z_k z_{k+1} + z_{k+1} z_{k+2}) \\ & + a(z_k z_{k+2} + z_{k+1}^2)] \end{aligned} \tag{38}$$

The estimation errors σs are

$$\begin{aligned} \sigma_a &= \frac{\sigma}{u_k z_{k+1} - u_{k+1} z_k} \sqrt{(1+\hat{a}^2)(u_k^2 + u_{k+1}^2} \\ \sigma_b &= \frac{\sigma}{u_k z_{k+1} - u_{k+1} z_k} \sqrt{(1+\hat{a}^2)(z_k^2 + z_{k+1}^2) + 2\hat{a} z_k z_{k+1}} \end{aligned} \tag{39}$$

It is appreciated that the source of difficulty in system identification is correlation. The latter is caused by measurement (sensor) noise, *not* process noise. Hence it is instructive to re-

consider the identification of this first-order control system in the case where a disturbance, namely process noise, enters the system and there is no measurement noise. As discussed in the next section, this yields a legitimate ARX model.

The dynamics are

$$y_{k+1} = ay_k + bu_k + w_k, \qquad k = 1, 2, \ldots \tag{40}$$

The process noise is a zero-mean Gaussian random variable with variance σ^2 and the process noise sequence is white, namely $E(w_k w_l) = 0$ for all $k \neq 1$. The measurement equation is

$$z_{k+1} = y_{k+1} \tag{41}$$

Since $y_k = z_k$ and $y_{k+1} = z_{k+1}$, the following holds:

$$z_{k+1} = az_k + bu_k + w_k \tag{42}$$

Hence the following linear regression is obtained:

$$\begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix} = \begin{bmatrix} z_k & u_k \\ z_{k+1} & u_{k+1} \\ \vdots & \vdots \\ z_{k+N-1} & u_{k+N-1} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} w_k \\ w_{k+1} \\ \vdots \\ w_{k+N-1} \end{bmatrix}$$

Define

$$Z = \begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix}, \quad H = \begin{bmatrix} z_k & u_k \\ z_{k+1} & u_{k+1} \\ \vdots & \vdots \\ z_{k+N-1} & u_{k+N-1} \end{bmatrix}, \quad W = \begin{bmatrix} w_k \\ w_{k+1} \\ \vdots \\ w_{k+N-1} \end{bmatrix}$$

Now

$$R = E(WW^{\mathrm{T}}) = \sigma^2 I_N \tag{43}$$

that is, the covariance matrix of the equation error is a scaled identity matrix. Hence the LS estimate is correct, since the parameter estimate

$$\begin{aligned} \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} &= (H^{\mathrm{T}}H)^{-1}H^{\mathrm{T}}Z \\ &= \begin{bmatrix} \sum_{i=1}^N z_{k+i-1}^2 & \sum_{i=1}^N z_{k+i-1}u_{k+i-1} \\ \sum_{i=1}^N z_{k+i-1}u_{k+i-1} & \sum_{i=1}^N u_{k+i-1}^2 \end{bmatrix}^{-1} \\ &\quad \begin{bmatrix} z_k & z_{k+1} & \cdots & z_{k+N-1} \\ u_k & u_{k+1} & \cdots & u_{k+N-1} \end{bmatrix} \begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N} \end{bmatrix} \end{aligned} \tag{44}$$

The explicit formulas for the LS parameter estimates are

$$\begin{aligned} \hat{a}_{LS} &= \frac{\sum_{i=1}^N u_{k+i-1}^2 \sum_{i=1}^N z_{k+i}z_{k+i-1} - \sum_{i=1}^N u_{k+i-1}z_{k+i-1} \sum_{i=1}^N z_{k+i}u_{k+i-1}}{\sum_{i=1}^N u_{k+i-1}^2 \sum_{i=1}^N z_{k+i-1}^2 - (\sum_{i=1}^N u_{k+i-1}z_{k+i-1})^2} \\ \hat{b}_{LS} &= \frac{\sum_{i=1}^N z_{k+i-1}^2 \sum_{i=1}^N z_{k+i}u_{k+i-1} - \sum_{i=1}^N u_{k+i-1}z_{k+i-1} \sum_{i=1}^N z_{k+i}z_{k+i-1}}{\sum_{i=1}^N u_{k+i-1}^2 \sum_{i=1}^N z_{k+i-1}^2 - (\sum_{i=1}^N u_{k+i-1}z_{k+i-1})^2} \end{aligned} \tag{45}$$

No iterations are required.

Moreover the estimation error covariance is

$$\begin{aligned} P &= \sigma^2(H^{\mathrm{T}}H)^{-1} \\ &= \sigma^2 \begin{bmatrix} \sum_{i=1}^N z_{k+i-1}^2 & \sum_{i=1}^N z_{k+i-1}u_{k+i-1} \\ \sum_{i=1}^N z_{k+i-1}u_{k+i-1} & \sum_{i=1}^N u_{k+i-1}^2 \end{bmatrix}^{-1} \end{aligned}$$

The σ estimation errors of the system's a and b parameter is

$$\begin{aligned} \sigma_a &= \sigma \frac{\sqrt{\sum_{i=1}^N u_{k+i-1}^2}}{\sqrt{\sum_{i=1}^N u_{k+i-1}^2 \sum_{i=1}^N z_{k+i-1}^2 - (\sum_{i=1}^N u_{k+i-1}z_{k+i-1})^2}} \\ \sigma_b &= \sigma \frac{\sqrt{\sum_{i=1}^N z_{k+i-1}^2}}{\sqrt{\sum_{i=1}^N u_{k+i-1}^2 \sum_{i=1}^N z_{k+i-1}^2 - (\sum_{i=1}^N u_{k+i-1}z_{k+i-1})^2}} \end{aligned} \tag{46}$$

Identification Experiment

Simulation experiments validate the above insights and derivation. The truth model's parameters are

$$\begin{aligned} a &= 0.95 \\ b &= 1 \end{aligned}$$

and the intensity of the *measurement* noise is determined by

$$\sigma = 0.1$$

The input signal is

$$u_k = \sin(0.1k), \qquad k = 0, 1, \ldots, 9$$

and the prior information is

$$\begin{aligned} \hat{a}_0 &= 0.8 \\ \hat{b}_0 &= 1.2 \end{aligned} \tag{47}$$

or

$$\begin{aligned} \hat{a}_0 &= 0.5 \\ \hat{b}_0 &= 1.5 \end{aligned} \tag{48}$$

The LS estimate is given by Eqs. (51) and (52):

$$\begin{aligned} \hat{a} &= 0.9390 \\ \hat{b} &= 1.0257 \end{aligned} \tag{49}$$

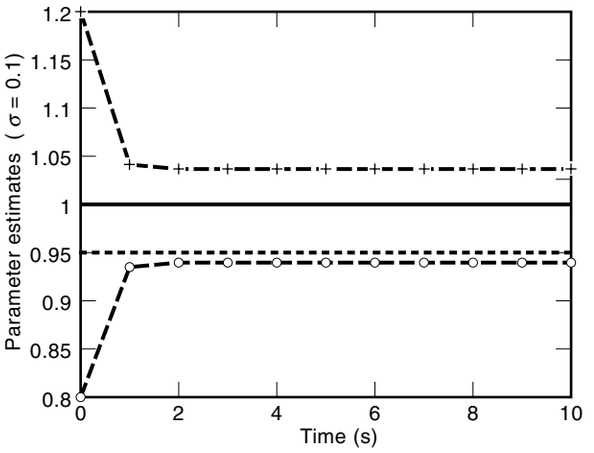

Figure 6. Iterated parameter estimates ($\hat{a}_0 = 0.8$, $\hat{b}_0 = 1.2$).

and the σ estimation errors of the system's a and b parameters are

$$\begin{aligned} \sigma_a &= 0.022 \\ \sigma_b &= 0.065 \end{aligned} \tag{50}$$

The MV parameter estimate is iteratively determined according to Eqs. (40) to (42). The (fast) convergence of the estimates is graphically illustrated in Figs. 6 and 7 for the prior information in Eqs. (53) and (54), respectively.

The identification results are summarized in Table 1.

The σ estimation errors are

$$\begin{aligned} \sigma_a &= 0.042 \\ \sigma_b &= 0.108 \end{aligned} \tag{51}$$

and

$$\begin{aligned} \sigma_a &= 0.05 \\ \sigma_b &= 0.14 \end{aligned} \tag{52}$$

for the prior information in Eqs. (53) and (54), respectively.

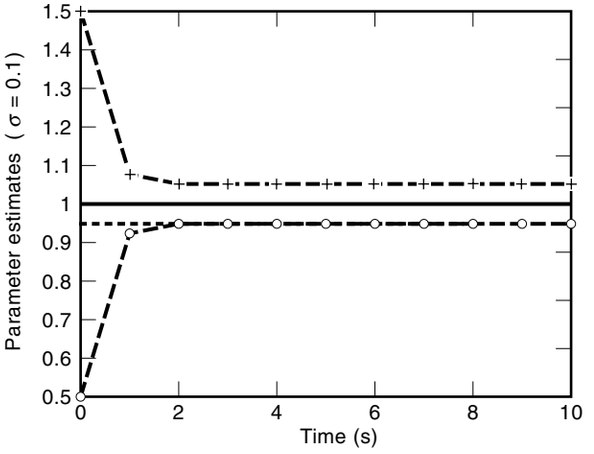

Figure 7. Iterated parameter estimates ($\hat{a}_0 = 0.5$, $\hat{b}_0 = 1.5$).

Table 1. Estimation Performance

$\hat{a}_0$	$\hat{b}_0$	$\hat{a}_1$	$\hat{b}_1$	$\hat{a}$	$\hat{b}$
0.8	1.2	0.9425	1.0177	0.9427	1.0173
0.5	1.5	0.9415	1.0200	0.9427	1.0173

The experimental results show the following:

1. The MV estimates are superior to the LS estimates.
2. The calculated confidence level in the MV estimates is lower than in the LS estimates.

According to item 1 above, the MV estimates are less biased than the LS estimates. Moreover, item 2 above suggests that the system identification scheme based on MV estimation is less prone to the notorious "divergence" of EKFs (extended Kalman filtering). Indeed it is reasonable to gauge the estimation performance of a system identification algorithm using the metric

$$\frac{|\hat{a} - a|}{\sigma_a} + \frac{|\hat{b} - b|}{\sigma_b}$$

In view of 1 and 2 above one concludes that the performance of the system identification scheme based on MV estimation is superior to the LS based system identification, as expected.

Flight Control Application

In this flight control application, system identification experiments concerning the pitch dynamics of an aircraft are presented [e.g., see (17,18)]. The identification of the "short-period" dynamics of an aircraft is considered first. The truth model is adapted from Ref. 16: At a certain flight condition the transfer function, which represents the aircraft's pitch rate y in response to elevator deflection u, is

$$\frac{y(s)}{u(s)} = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2} = \frac{4.8s + 1.44}{s^2 + 0.84s + 1.44}$$

The Bode plot for this transfer function, which is used for innerloop flight control system design, is shown in Fig. 8. Rep-

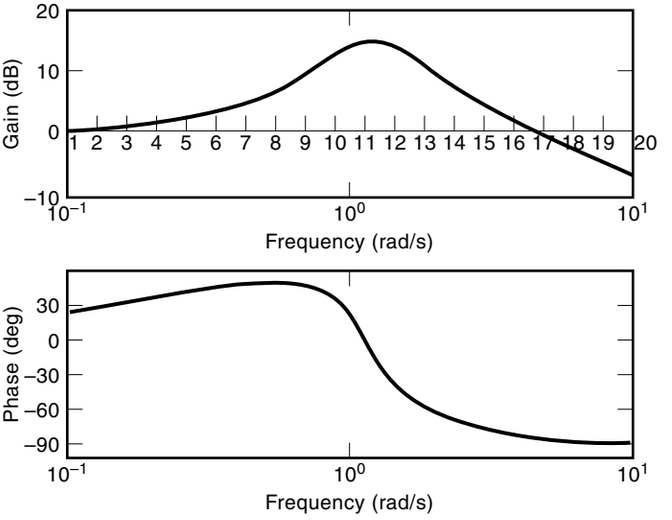

Figure 8. Second-order Bode plot.

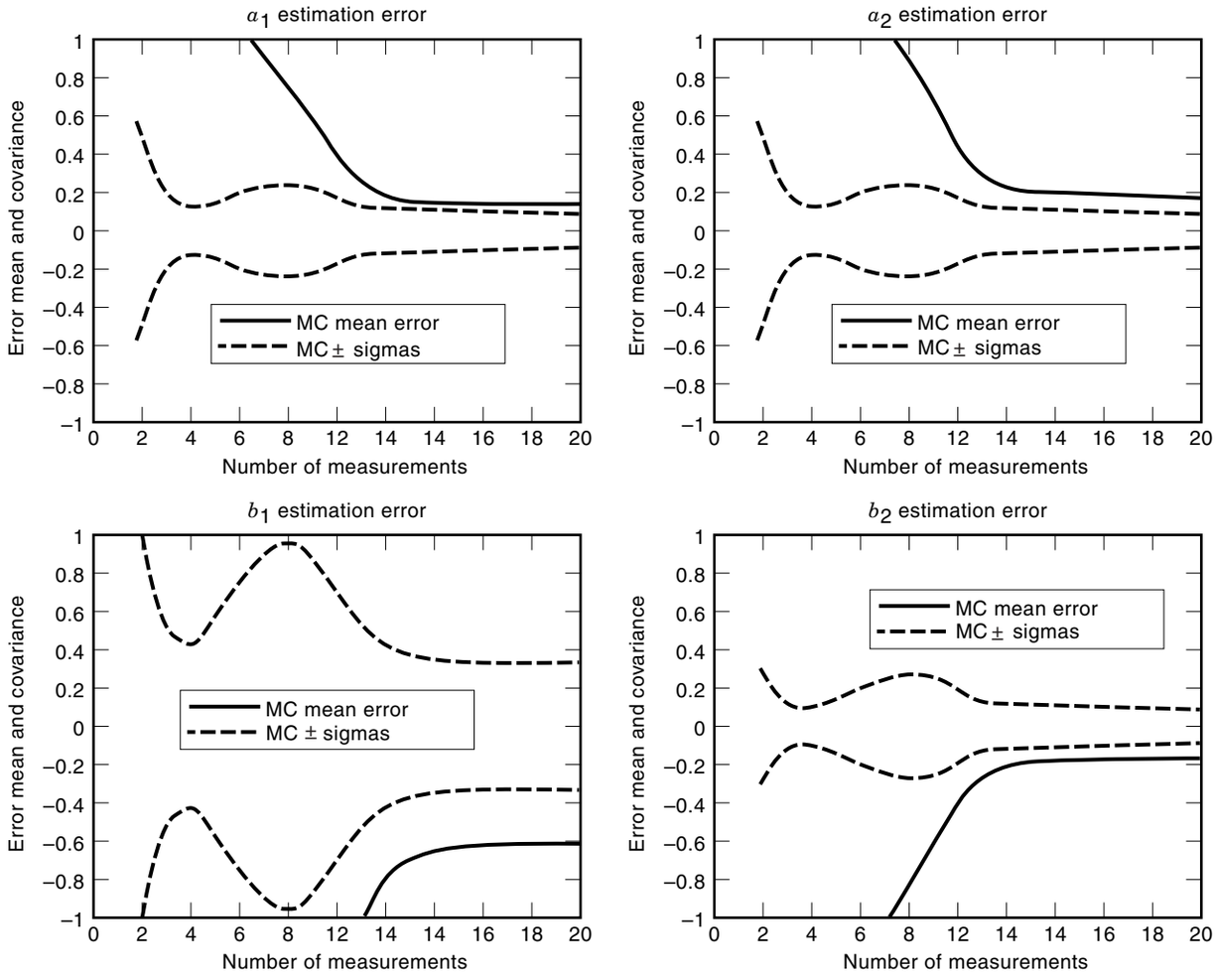

Figure 9. LS estimate.

resentative measurement noise values for use in this example are representative of a Tektronix frequency analyzer. The manufacturer's specifications give measurement error values of ± 0.2 dB and $\pm 0.5°$. These are taken conservatively as two sigma values for the noise on the amplitude and phase measurements.

If the LS estimate is used, then the results in Fig. 9 are obtained. Shown in the plots are average error and estimation error covariances, which are the result of a 20-run Monte Carlo experiment and which, on evoking the weak law of large numbers, render a gauge of the identification algorithm's estimation bias. As can be seen in the plots, there are large biases in the average errors, which are outside the one sigma bounds. This implies that the majority of the parameter estimates in the 20 runs are outside this bound.

However, when the correct MV estimate is used, the results improve dramatically, as Fig. 10 shows. The biases and estimation error covariances are much smaller (note the scale difference). Included in these plots is the system identification algorithm predicted estimation error covariance. The predicted estimation error covariance is very close to the realized estimation error covariance after about nine (phasor) measurements. The small biases and covariance differences imply that the MV system identification algorithm yields relatively unbiased parameter estimates. Additionally the novel MV system identification algorithm is doing a good job of predicting the accuracy of its estimate; that is to say, the algorithm is efficient.

The results of the experiments are summarized in Table 2. In all cases the values shown correspond to a 16 (phasor) measurement linear regression. The Monte Carlo averaged estimation error (e) and sigma (σ_e) are given for the LS and MV algorithms, and the algorithm predicted sigma (σ_p) is given for the MV experiments. In each case the bias in the MV estimates is about two orders of magnitude smaller than in the naive LS estimate, and the estimation error covariance is much smaller as well. Finally, when Bode plots are constructed using $\theta + e \pm 2\sigma$ as parameter values, all the plots lie virtually on top of one another.

In sum, the modeling of a realistic process of system identification using a frequency analyzer instrument takes a frequency domain approach, and phasors are used for system identification. Gaussian measurement noise is assumed, as is customary in classical filtering and system identification work. The proper minimum-variance estimate equations are derived and compared with the simple least-squares estimate

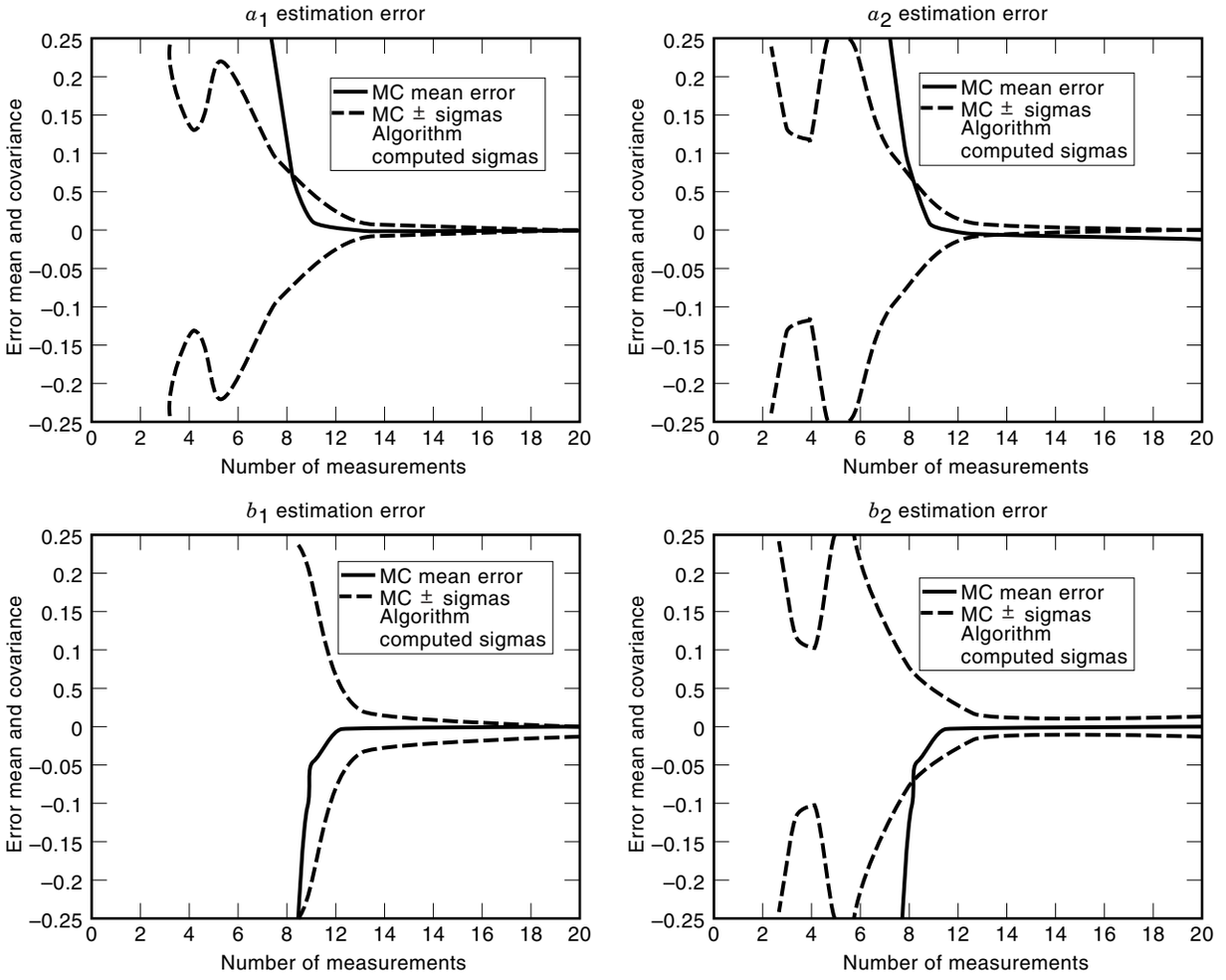

Figure 10. MV estimate.

for a second-order dynamical system representative of an aircraft's pitch channel, and this is used for innerloop flight control system design.

While the parameter vector estimate is initially very poor for just a few (phasor) measurements, the estimation error covariance converges quickly as more measurements are added until a steady state is achieved at around 16 measurements. For this reason 16 measurements are taken in the iterated MV estimation. In all cases the MV estimate outperformed the LS estimate.

Table 2. Constant Strength Noise on $R_{k\text{dB}}$ and $\phi_{k\text{deg}}$

Estimate	θ	$E = \hat{\theta} - \theta$	σ_e	σ_{p}
Least	a_1	2.527×10^{-2}	3.319×10^{-2}	NA
mean	a_2	3.542×10^{-2}	2.666×10^{-2}	NA
squares	b_1	-1.210×10^{-1}	9.390×10^{-2}	NA
	b_2	-3.364×10^{-2}	3.132×10^{-2}	NA
Twice	a_1	-1.531×10^{-3}	2.951×10^{-3}	3.162×10^{-3}
iterated	a_2	-7.473×10^{-4}	1.820×10^{-3}	3.208×10^{-3}
Minimum	b_1	1.994×10^{-3}	1.318×10^{-2}	1.616×10^{-2}
variance	b_2	8.456×10^{-4}	2.884×10^{-3}	6.487×10^{-3}

A recent application of the iterated MV system identification algorithm to deep-level transient spectroscopy experiments in physics is documented in Ref. 19.

CONCLUSION

Multidimensional signal processing and system identification are the mainstay of signal processing. They address the respective deterministic and stochastic aspects and should be synergistically applied. The presentation of this article while very basic is not elementary, and a wealth of extensions and applications is possible.

BIBLIOGRAPHY

1. D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing,* Englewood Cliffs, NJ: Prentice-Hall, 1984.
2. J. S. Lim, *Two-Dimensional Signal and Image Processing,* Englewood Cliffs, NJ: Prentice-Hall, 1990.
3. B. W. Suter, *Multirate and Wavelet Signal Processing,* San Diego, CA: Academic Press, 1997.

4. J. W. S. Cassels, *An Introduction to the Geometry of Numbers,* New York: Springer-Verlag, 1959.
5. D. P. Petersen and D. Middleton, Sampling and reconstruction of wave-numbered functions in *N*-dimensional Euclidean space, *Inf. Control,* **5**: 279–323, 1962.
6. E. Viscito and J. P. Allebach, The analysis and design of multidimensional FIR perfect reconstruction filter banks, *IEEE Trans. Circuits Syst.,* **38**: 29–41, 1991.
7. L. Ljung, *System Identification for the User,* Englewood Cliffs, NJ: Prentice-Hall, 1987.
8. P. Eykhoff and P. C. Parks, Identification and system parameter estimation—Where do we stand now?, *Automatica,* **26** (1): 3–5, 1990.
9. M. Pachter and P. R. Chandler, Universal linearization concept for extended Kalman filters, *IEEE Trans. Aerosp. Electron. Syst.,* **29**: 946–961, 1993.
10. D. A. Belsley et al., *Regression Diagnostics,* New York: Wiley, 1990.
11. P. R. Chandler, M. Pachter, and M. J. Mears, Constrained linear regression for flight control system failure identification, *Proc. Amer. Control Conf.,* San Francisco, CA: 1993, pp. 3141–3144.
12. P. R. Chandler, M. Pachter, and M. Mears, On line optimizing networks for reconfigurable control, *Proc. 1993 Conf. Decision Control,* San Antonio, TX, 1993, pp. 2272–2275.
13. M. J. Mears et al., A Hopfield neural network for adaptive control, *Proc. 1993 AIAA Conf. Guidance, Navigation Control,* Monterey, CA, 1993, pp. 276–284.
14. P. Chandler, M. Pachter, and M. Mears, System identification for adaptive and reconfigurable control, *AIAA J. Guidance, Control, Dynamics,* **18** (3): 516–524, 1995.
15. W. L. Brogan, *Modern Control Theory,* New York: Quantum, 1974, p. 48.
16. J. H. Blakelock, *Automatic Control of Aircraft and Missiles,* New York: Wiley, 1991.
17. O. Reynolds and M. Pachter, System identification algorithm using phasors, *Eur. Control Conf.,* Brussels, Belgium, 1997.
18. O. Reynolds and M. Pachter, The performance of generalized minimum Variance System Identification, *5th IEEE Mediterranean Conf. Control Syst.,* Paphos, Cyprus, 1997.
19. E. A. Ingham, J. D. Scofield, and M. Pachter, Improved linear prediction for deep level transient spectroscopy analysis, *J. Appl. Phys.,* **80** (5): 2805–2814, 1996.
20. L. Smith, P. R. Chandler, and M. Pachter, Regularization techniques for real-time identification of aircraft parameters, *Proc. 1997 AIAA Guidance, Navigation Control Conf.,* AIAA paper 97-3740, New Orleans, LA: 1997, pp. 1466–1480.

B. SUTER
Air Force Research Laboratory
(AFRL/IFGC)

M. PACHTER
Air Force Institute of Technology
(AFIT/ENG)

MULTIFUNCTION PARALLELISM. See PARALLEL PROCESSING, SUPERSCALAR AND VLIW PROCESSORS.

MULTIMEDIA. See HYPERMEDIA.