CHAPTER 9

REALIZATION AND SUBSPACE IDENTIFICATION

Topics covered

- State-Space Realization of Transfer Function
- Minimal Realization (MR)
- Subspace Identifications (Time and Frequency Domain)

9.1 INTRODUCTION

In this chapter, we consider the problems of state-space **realization** and **identification**.

The state-space realization problem is the problem to find the matrices A, B, C, and D of the transfer function $G(s)$ in the continuous-time case or $G(z)$ in the discrete-time case, given a set of large number of Markov parameters.

In case of a discrete-time system, the Markov parameters can easily be computed from the input–output sequence of the systems (see **Exercise 9.5**). Finding Markov parameters in the case of a continuous-time system is not that straightforward.

There may exist many realizations of the same transfer function matrix. Two such realizations, **controllable and observable realizations**, are obtained in **Section 9.2.1**.

A realization with the smallest possible dimension of A is called a **minimal realization** (MR). A necessary and sufficient condition for a realization to be an MR is established in **Theorem 9.2.1**, and it is shown in **Theorem 9.2.2** that two MRs are related via a nonsingular transformation.

The existing algorithms for finding MRs are all based on factoring the associated Hankel matrix (matrices) of Markov parameters. Some basic rank properties of these matrices, which are relevant to such factorizations, are established in **Section 9.3**.

Two **numerically viable algorithms (Algorithms 9.3.1 and 9.3.2)** based on the **singular value decomposition(s)** (SVD) of these matrices are then described in Section 9.3. The algorithms are valid both for continuous-time and discrete-time state-space realizations, provided the Markov parameters are known.

The identification problem is the problem of identifying system matrices A, B, C, and D from a given set of input–output data, rather than Markov parameters.

Two time-domain subspace system identification algorithms (**Algorithms 9.4.1 and 9.4.2**) are presented in **Section 9.4.** These algorithms are based on the SVD decompositions of Hankel matrices constructed directly from the input–output sequences. The algorithms are presented for discrete-time systems identification, but can be used for identifying the continuous-time systems also, provided the first and higher derivatives of the inputs and outputs can be computed. In the last section (**Section 9.4.4**), we state a **frequency-domain subspace identification** algorithm (**Algorithm 9.4.3**). A frequency-domain state-space identification is concerned with finding the system matrices, given a set of measured frequency responses. The algorithm is stated for identification of a continuous-time system; however, it can be used for discrete-time identification also, with trivial modifications.

Reader's Guide

The readers familiar with material on state-space realization can skip Sections 9.2 and 9.3.1.

9.2 STATE-SPACE REALIZATIONS OF A TRANSFER FUNCTION

In this section, we show, given a transfer matrix, how to construct state-space realizations in controllable and observable forms of this transfer matrix.

We consider here only the continuous-time case. The results are also valid for the discrete-time case by replacing the variable s by the variable z.

Definition 9.2.1. *Let $G(s)$ be the transfer matrix of order $r \times m$ which is proper. Then the quadruple (A, B, C, D) such that*

$$G(s) = C(sI - A)^{-1}B + D \tag{9.2.1}$$

is called a **state-space realization** *of $G(s)$.*

It can be shown (**Exercise 9.1**) that given a proper rational function $G(s)$, there always exists a state-space realization of $G(s)$. However, **such a realization is not unique**, that is, there may exist many state-space realizations of the same transfer matrix.

In the following sections we show the non-uniqueness of the state-space realization of a transfer matrix (for the single-input, single-output case (SISO)), by constructing two realizations of the same transfer matrix.

9.2.1 Controllable and Observable Realizations

The transfer matrix $G(s)$ can be written in the form:

$$G(s) = D + \frac{P(s)}{d(s)}, \tag{9.2.2}$$

where $P(s)$ is a polynomial matrix in s of degree at most $h-1$ given by

$$P(s) = P_0 + P_1 s + \cdots + P_{h-1} s^{h-1}, \tag{9.2.3}$$

and $d(s) = s^h + d_{h-1}s^{h-1} + \cdots + d_1 s + d_0$ is a monic polynomial of degree h (h is the least common multiple of the denominators of all the entries of $G(s)$).

Let 0_p and I_p denote, respectively, the zero and identity matrices of order p.

Define now

$$A = \begin{pmatrix} 0_m & I_m & & & \\ & 0_m & I_m & & \\ \vdots & & \ddots & \ddots & \\ 0_m & & \cdots & 0_m & I_m \\ -d_0 I_m & -d_1 I_m & -d_2 I_m & \cdots & -d_{h-1} I_m \end{pmatrix}, \tag{9.2.4}$$

$$B = \begin{pmatrix} 0_m \\ 0_m \\ \vdots \\ 0_m \\ I_m \end{pmatrix}, \qquad C = (P_0, \ldots, P_{h-1}), \tag{9.2.5}$$

$$D = \lim_{s \to \infty} G(s), \tag{9.2.6}$$

Then it is easily verified that

$$C(sI - A)^{-1}B + D = G(s) = D + \frac{P(s)}{d(s)}. \tag{9.2.7}$$

Since the matrix-pair (A, B) is controllable, the above realization of $G(s)$ is called a **controllable realization.** This realization has dimension mh.

We now construct a different realization of $G(s)$.

Expand $G(s)$ in Taylor series:

$$G(s) = D' + \frac{1}{s}H_1 + \frac{1}{s^2}H_2 + \cdots \tag{9.2.8}$$

The matrices $\{D', H_k\}$ can be found as follows:

$$\begin{aligned} D' &= \lim_{s\to\infty} G(s) \\ H_1 &= \lim_{s\to\infty} s(G(s) - D') \\ H_2 &= \lim_{s\to\infty} s^2\left(G(s) - D' - \frac{1}{s}H_1\right) \\ &\vdots \\ &\text{etc.} \end{aligned} \tag{9.2.9}$$

Definition 9.2.2. *The matrices $\{H_i\}$, defined above, are called the* **Markov parameters** *of $G(s)$.*

Note: The Markov parameters $\{H_i\}$ can be expressed as:

$$H_i = CA^{i-1}B, \quad i = 1, 2, \ldots \tag{9.2.10}$$

Define now the matrices A', B', and C' as follows:

$$A' = \begin{pmatrix} 0_r & I_r & & & \\ & 0_r & I_r & & \\ \vdots & & \ddots & \ddots & \\ 0_r & & \cdots & 0_r & I_r \\ -d_0 I_r & -d_1 I_r & -d_2 I_r & \cdots & -d_{h-1} I_r \end{pmatrix}, \tag{9.2.11}$$

$$B' = \begin{pmatrix} H_1 \\ H_2 \\ H_3 \\ \vdots \\ H_h \end{pmatrix}, \qquad C' = (I_r,\ 0_r,\ \ldots,\ 0_r)\,. \tag{9.2.12}$$

Then it can be shown that with A', B', C', and D' as defined above, we have

$$C'(sI - A')^{-1}B' + D' = G(s). \tag{9.2.13}$$

That is, we have now another realization of $G(s)$. Since (A', C') is observable, this realization is called an **observable realization** of $G(s)$. This realization has dimension rh.

9.2.2 Minimal Realization

Since there may exist more than one realization of the same transfer function $G(s)$, it is natural to look for a realization of minimal order.

Definition 9.2.3. *A state-space realization (A, B, C, D) of $G(s)$ is said to be an MR of $G(s)$ if the matrix A has the smallest possible dimension, that is, if (A', B', C', D') is any other realization of $G(s)$, then the order of A' is greater than or equal to the order of A. The dimension of an MR is called the* ***McMillan degree.***

Theorem 9.2.1. *A state-space realization (A, B, C, D) of $G(s)$ is minimal if and only if (A, B) is controllable and (A, C) is observable.*

Proof. We first prove the necessity by **contradiction.**

If (A, B) is not controllable and/or (A, C) is not observable, then from Kalman decomposition **(see Chapter 6)**, it follows that there exists a realization of smaller dimension that is both controllable and observable. This contradicts the minimality assumption.

Conversely, let (A, B, C, D) and (A', B', C', D') be two minimal realizations of $G(s)$. Assume that the order of A' is $n' < n$. Since the two realizations have the same transfer function, then they should have the same Markov parameters, that is,

$$CA^{i-1}B = C'(A')^{i-1}B'. \tag{9.2.14}$$

This implies that

$$O_{\mathrm{M}}C_{\mathrm{M}} = O'_{\mathrm{M}}C'_{\mathrm{M}}, \tag{9.2.15}$$

where O_{M} and C_{M}, respectively, denote the observability and controllability matrices of the realization (A, B, C, D) and, O'_{M} and C'_{M}, respectively, denote the observability and controllability matrices of the realization (A', B', C', D').

But, $\operatorname{rank}(O_{\mathrm{M}}C_{\mathrm{M}}) = n$, and $\operatorname{rank}(O'_{\mathrm{M}}C'_{\mathrm{M}}) = n' < n$. This is a contradiction, since $\operatorname{rank}(O_{\mathrm{M}}C_{\mathrm{M}}) = \operatorname{rank}(O'_{\mathrm{M}}C'_{\mathrm{M}})$, by (9.2.15). ■

The next question is how are two MRs of the same transfer matrices related? We answer the question in Theorem 9.2.2.

Theorem 9.2.2. *If (A, B, C, D) and (A', B', C', D') are two MRs of the same transfer function $G(s)$, then there exists a unique nonsingular matrix T such that*

$$A' = T^{-1}AT, \tag{9.2.16}$$

$$B' = T^{-1}B, \qquad C' = CT, \qquad D' = D. \tag{9.2.17}$$

Moreover, T is explicitly given by

$$T = \left(O_{\mathrm{M}}^{\mathrm{T}}O_{\mathrm{M}}\right)^{-1} \cdot O_{\mathrm{M}}^{\mathrm{T}}O'_{\mathrm{M}} \tag{9.2.18}$$

or

$$T = C_{\mathrm{M}}(C'_{\mathrm{M}})^{\mathrm{T}}[C'_{\mathrm{M}}(C'_{\mathrm{M}})^{\mathrm{T}}]^{-1}, \tag{9.2.19}$$

where C_M and O_M are, respectively, the controllability and observability matrices of the realization (A, B, C, D), and C'_M and O'_M are, respectively, the controllability and observability matrices of the realization (A', B', C', D').

Proof. We just sketch a proof here and leave the details to the readers.

Let T be the matrix relating the matrices O_M and O'_M, that is, T satisfies the matrix equation:

$$O_M T = O'_M. \tag{9.2.20}$$

Since O_M has full rank, such a matrix T always exists. In fact, it is unique and is given by

$$T = (O_M^T O_M)^{-1} O_M^T O'_M. \tag{9.2.21}$$

From the first block row of Eq. (9.2.20), we have $CT = C'$.

Since both the realizations have the same transfer function, and hence the same Markov parameters, we obtain

$$O_M C_M = O'_M C'_M, \tag{9.2.22}$$

which gives

$$C_M = \left(O_M^T O_M\right)^{-1} O_M^T O'_M C'_M = T C'_M. \tag{9.2.23}$$

That is, T is a solution of the equation

$$T C'_M = C_M. \tag{9.2.24}$$

Since C'_M has full rank, we have

$$T = C_M (C'_M)^T [C'_M (C'_M)^T]^{-1}, \text{ establishing (9.2.19).}$$

Again, from the first block column of Eq. (9.2.23), we have

$$T B' = B. \tag{9.2.25}$$

All that remains to be shown is that (9.2.16) holds. To show this, first note that the Markov parameters $CA^{i-1}B$ and $C'(A')^{i-1}B'$, $i \geq 1$, are equal.

We can then write

$$O_M A C_M = O'_M A' C'_M, \tag{9.2.26}$$

which leads to

$$O_M^T O_M A C_M = O_M^T O'_M A' C'_M. \tag{9.2.27}$$

From (9.2.27) we have

$$AC_M = TA'C'_M \text{ (where } T \text{ is defined by (9.2.18)).} \tag{9.2.28}$$

But again multiplying (9.2.19) by A to the left, we have

$$AC_M(C'_M)^T(C'_M(C'_M)^T)^{-1} = AT. \tag{9.2.29}$$

From (9.2.28) and (9.2.29), we obtain

$$AT = TA'$$

That is, $A' = T^{-1}AT$. ■

Uniqueness: Suppose that there exists another similarity transformation given by T_1 relating both the systems. Then we must have:

$$O_M(T - T_1) = 0.$$

But O_M has full rank, so, $T = T_1$.

Example 9.2.1. Let

$$G(s) = \frac{3s - 4}{s^2 - 3s + 2}.$$

Here

$$P(s) = -4 + 3s,$$
$$d(s) = s^2 - 3s + 2.$$

The Markov parameters are:

$$D' = \lim_{s\to\infty} G(s) = 0,$$
$$H_1 = \lim_{s\to\infty} s(G(s) - D') = 3,$$
$$H_2 = \lim_{s\to\infty} s^2\left(G(s) - D' - \frac{1}{s}H_1\right) = 5.$$

(I) *Controllable realization*

$$A = \begin{pmatrix} 0 & 1 \\ -2 & 3 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad C = (-4, 3), \qquad D = 0.$$

Verify:

$$C(sI - A)^{-1}B = \frac{1}{s^2 - 3s + 2}(-4, 3)\begin{pmatrix} s-3 & 1 \\ -2 & s \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$
$$= \frac{3s - 4}{s^2 - 3s + 2}.$$

Since (A, B) is controllable and (A, C) is observable, the realization is an **MR**.

(II) *Observable realization*

$$A' = \begin{pmatrix} 0 & 1 \\ -2 & 3 \end{pmatrix}, \qquad B' = \begin{pmatrix} 3 \\ 5 \end{pmatrix}, \qquad C' = (1, 0).$$

Verify: $C'(sI - A)^{-1}B' = \dfrac{3s - 4}{s^2 - 3s + 2}$.

Since (A', B') is controllable, and (A', C') is observable, this is also an MR of $G(s)$.

(III) *Relationship.* The two realizations are related by the nonsingular transforming matrix T given by

$$T = (O_M^T O_M)^{-1} O_M^T O_M' = \begin{pmatrix} -2.5 & 1.5 \\ -3 & 2 \end{pmatrix}.$$

Verify: $T^{-1}AT = \begin{pmatrix} 0 & 1 \\ -2 & 3 \end{pmatrix} A'$, $T^{-1}B = \begin{pmatrix} 3 \\ 5 \end{pmatrix} = B'$, $CT = (1, 0) = C'$.

9.3 COMPUTING MINIMAL REALIZATIONS FROM MARKOV PARAMETERS

In the last section, we showed how to obtain an observable realization from a set of Markov parameters:

$$H_k = CA^{k-1}B, \quad k = 1, 2, \ldots$$

Here we consider the problem of computing a MR using Markov parameters. Specifically, the following problem is considered.

> Given a set of large number Markov parameters $\{H_k\}$ of an unknown transfer function $G(s)$, find a minimal realization (A, B, C, D) whose transfer function $G(s) = C(sI - A)^{-1}B + D$.

Since the Markov parameters are much easier to obtain for a discrete-time system, unless otherwise stated, we assume that the given Markov parameters are of the discrete-time system:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k, \\ y_k &= Cx_k + Du_k. \end{aligned} \tag{9.3.1}$$

9.3.1 Some Basic Properties of the Hankel Matrix of Markov Parameters

There exist many methods for finding a minimal realization (see DeJong (1978) for a survey). Most of these methods find a minimal realization from a decomposition or a factorization of the Hankel matrix of Markov parameters of the form:

$$M_k = \begin{pmatrix} H_1 & H_2 & \cdots & H_k \\ H_2 & H_3 & \cdots & H_{k+1} \\ \vdots & \vdots & & \vdots \\ H_k & H_{k+1} & \cdots & H_{2k-1} \end{pmatrix}. \tag{9.3.2}$$

For example, a recursive method due to Rissanen (1971) obtains a minimal realization by recursively updating the decomposition of a smaller Hankel matrix to that of a larger Hankel matrix.

The following basic results due to Kalman *et al.* (see, e.g., Kalman *et al.* (1969), play an important role in the developments of Rissanen's and other methods.

Theorem 9.3.1.

(i) $\text{Rank}(M_k) \leq \text{rank}(M_{k+1})$ *for all* k.

(ii) *If* (A, B, C, D) *is any realization of dimension* n, *then* $\text{rank}(M_k) = \text{rank}(M_n)$ *for all* $k \geq n$.

(iii) *Let* (A, B, C, D) *and* (A', B', C', D') *be two realizations of* $G(s)$ *of order* n *and* n', *respectively. Then,*

$$\text{rank}(M_n) = \text{rank}(M_n').$$

(iv) *Let* d *be the McMillan degree, then* $\max_k(\text{rank}(M_k)) = d$.

(v) *Let* (A, B, C, D) *be any realization of dimension* n, *then*

$$d = \text{rank}(M_n) = \text{rank}(O_M C_M),$$

where O_M *and* C_M *are, respectively, the observability and controllability matrices of the realization* (A, B, C, D).

Proof.

(i) The proof of (i) follows from the fact that M_k is a submatrix of M_{k+1}.

(ii) The proof of (ii) follows by observing that (**Exercise 9.6**) the Hankel matrix M_k can be decomposed as:

$$M_k = \begin{pmatrix} O_M \\ \hline CA^n \\ \vdots \\ CA^{k-1} \end{pmatrix} \left(C_M | A^n B, A^{n+1} B, \ldots, A^{k-1} B) \right), \tag{9.3.3}$$

$$\text{for } k \geq n \text{ and } M_n = O_M C_M. \tag{9.3.4}$$

Since the rows in $(CA^n, \ldots, CA^{k-1})$ and the columns in $(A^n B, \ldots, A^{k-1} B)$ are linear combination of the rows in O_M and the columns in C_M, respectively, we have

$$\text{rank}(M_k) = \text{rank}(M_n) = \text{rank}(O_M C_M).$$

(iii) Let (A', B', C', D') be another realization of $G(s)$ of order n' and let $r = \max(n, n')$. Then, since both these realization have the same Markov parameters, we must have

$$M_r' = M_r.$$

Thus by (ii), $\text{rank}(M_n) = \text{rank}(M_r) = \text{rank}(M_r') = \text{rank}(M_n')$.

(iv) The proof is by *contradiction*. Suppose that there exists a minimal realization (A, B, C, D) of order $d' < d$.

Then by the previous two results, we should have $\max((\text{rank}(M_k))) = d_1 < d'$, a contradiction.

(v) The proof follows from (iii) and (iv). ■

Finding the McMillan Degree

The above result gives us several alternative procedures to obtain the McMillan degree of the transfer function matrix.

A simple way to do so is to find any realization of $G(s)$ and then find the rank of the product $O_M C_M$, using the SVD.

Also, if the realization is stable and C_G and O_G are, respectively, the controllability and observability Grammians obtained via solutions of respective Lyapunov equations (**see Chapter 7**), then it is well known (Glover 1984) that the McMillan degree is equal to the rank of $C_G O_G$.

9.3.2 An SVD Method for Minimal Realization

It was shown by DeJong (1978) that the **Rissanen's method is numerically unstable**.

Since the SVD provides a numerically reliable way to compute the rank of a matrix, a more numerically viable method for finding an MR should be based on the SVD of the associated Hankel matrix. We now describe below an SVD-based method for finding an MR (Ho and Kalman 1966; Zeiger and McEwen 1974; Kung 1978). **For the sake of convenience, we will assume that $D = 0$ in this section.**

Given the set $\{H_1, H_2, \ldots, H_{2N+1}\}$ of Markov parameters, consider the SVD of M_{N+1}:

$$M_{N+1} = USV^{\mathrm{T}} = US^{1/2}S^{1/2}V^{\mathrm{T}} = U'V',$$

where $S = \mathrm{diag}(s_1, s_2, \ldots, s_p, 0, \ldots, 0)$, $U' = US^{1/2}$, and $V' = S^{1/2}V^{\mathrm{T}}$.

Comparing this decomposition with the decomposition of M_{N+1} in the form (9.3.2) in Section 9.3.1, it is easy to see that we can take C as the first block row and the first p columns of U' and similarly B can be taken as the first p rows and the first block column of V'.

The matrix A satisfies the relations:

$$U_1 A = U_2 \quad \text{and} \quad AV_1 = V_2,$$

where
$U_1 =$ The first N block rows and the first p columns of U'
$V_1 =$ The first p rows and the first N block columns of V'.
U_2 and V_2 are similarly defined. Since U_1 and V_1 have full ranks, we immediately have from the above two equations,

$$A = U_1^{\dagger}U_2 \quad \text{or} \quad A = V_2V_1^{\dagger},$$

where $U_1^{\dagger}$ and $V_1^{\dagger}$ are the generalized inverses of U_1 and V_1, respectively.

This discussion leads to the following SVD algorithm for finding an **MR**:

Algorithm 9.3.1. *An SVD Algorithm for Minimal Realization*

Inputs. *The set of Markov parameters:* $\{H_1, H_2, \ldots, H_{2N+1}\}$ *(N should be at least equal to the McMillan degree).*

Outputs. *The matrices A, B, and C of a minimal realization.*

Step 1. *Find the SVD of the block Hankel matrix*

$$M_{N+1} = \begin{pmatrix} H_1 & H_2 & \cdots & H_{N+1} \\ H_2 & H_3 & \cdots & H_{N+2} \\ \vdots & & & \\ H_{N+1} & H_{N+2} & \cdots & H_{2N+1} \end{pmatrix} = USV^{\mathrm{T}},$$

where $S = \mathrm{diag}(s_1, s_2, \ldots, s_p, 0, \ldots, 0)$, *and* $s_1 \geq s_2 \geq \cdots \geq s_p > 0$

Step 2. *Form* $U' = US^{1/2}$ *and* $V' = S^{1/2}V^{\mathrm{T}}$,
where $S^{1/2} = \mathrm{diag}(s_1^{1/2}, s_2^{1/2}, \ldots, s_p^{1/2}, 0, \ldots, 0)$.

Step 3. *Define*

U_1 = *The first N block rows and the first p columns of* U'

U_2 = *The last N block rows and the first p columns of* U'

$U^{(1)}$ = *The first block row and the first p columns of* U'

$V^{(1)}$ = *The first p rows and the first block column of* V'.

Step 4. *Compute* $A = U_1^{\dagger} U_2$, *Set* $B = V^{(1)}$, $C = U^{(1)}$.

Theorem 9.3.2 proved by Kung (1978) shows that the MR obtained by Algorithm 9.3.1 enjoys certain desirable properties.

Theorem 9.3.2. *Let* E_i *denote the error matrix, that is,*

$$E_i = CA^{i-1}B - H_i, \quad i \geq 1.$$

Assume that the given impulse response sequence $\{H_k\}$ *is convergent. That is,* $H_k \to 0$, *when* $k \to \infty$.

Then,

- $\sum_{i=1}^{2N+1} \|E_i\|_F^2 \leq \epsilon\sqrt{n+m+r}$, *where* ϵ *is a small positive number, and* n, m *and* r *are, respectively, the number of states, inputs, and outputs.*
- *The minimal realization obtained by Algorithm* 9.3.1 *is (i)* **discrete-stable** *and (ii)* **internally balanced**, *that is, the controllability and observability Grammians for this realization are the same and are equal to a diagonal matrix* (*see* **Chapter 14**).

Example 9.3.1. Let $N = 2$ and the given set of Markov parameters be:

$$\{H_1, H_2, H_3, H_4, H_5\} = \{3, 5, 9, 17, 33\}.$$

Step 1. $M_3 = \begin{pmatrix} 3 & 5 & 9 \\ 5 & 9 & 17 \\ 9 & 17 & 33 \end{pmatrix}$. Then, $U = \begin{pmatrix} 0.2414 & -0.8099 & 0.5345 \\ 0.4479 & -0.3956 & -0.8018 \\ 0.8609 & 0.4330 & 0.2673 \end{pmatrix}$,

$S = \text{diag}(44.3689 \;\; 0.6311 \;\; 0)$, and $V^{\mathrm{T}} = \begin{pmatrix} 0.2414 & 0.4479 & 0.8609 \\ -0.8099 & -0.3956 & 0.4330 \\ 0.5345 & -0.8018 & 0.2673 \end{pmatrix}$.

Step 2.

$$U' = \begin{pmatrix} 1.6081 & -0.64340 & 0 \\ 2.9835 & -0.31430 & 0 \\ 5.7343 & 0.34400 & 0 \end{pmatrix},$$

$$V' = \begin{pmatrix} 1.6081 & 2.9835 & 5.7343 \\ -0.6434 & -0.3143 & 0.3440 \\ 0 & 0 & 0 \end{pmatrix}.$$

Step 3.

$$U_1 = \begin{pmatrix} 1.6081 & -0.6434 \\ 2.9835 & -0.3143 \end{pmatrix},$$

$$U_2 = \begin{pmatrix} 2.9835 & -0.3143 \\ 5.7343 & 0.3440 \end{pmatrix}.$$

$$U^{(1)} = (1.6081 \quad -0.6434),$$

$$V^{(1)} = \begin{pmatrix} 1.6081 \\ -0.6434 \end{pmatrix}.$$

Step 4.

$$A = U_1^{\dagger} U_2 = \begin{pmatrix} 1.9458 & 0.2263 \\ 0.2263 & 1.0542 \end{pmatrix},$$

$$B = V^{(1)} = \begin{pmatrix} 1.6081 \\ -0.6434 \end{pmatrix},$$

$$C = U^{(1)} = (1.6081 \quad -0.6434).$$

Verify:

$$E_1 = CB - H_1 = -8.8818 \times 10^{-16},$$

$$E_2 = CAB - H_2 = -8.8818 \times 10^{-16},$$

$$E_3 = CA^2B - H_3 = -1.7764 \times 10^{-15},$$

$$E_4 = CA^3B - H_4 = 0,$$

$$E_5 = CA^4B - H_5 = 7.1054 \times 10^{-15},$$

$$\sum_{i=1}^{5} |E_i|^2 = 5.5220 \times 10^{-30}.$$

It is also easy to check that the realization is both controllable and observable. So, it is minimal. The controllability and observability Grammians are the same and are given by: $C_{\mathrm{G}}^N = O_{\mathrm{G}}^N = \mathrm{diag}(44.3689,\ 0.6311)$.

Figure 9.1 shows a comparison between the graphs of the transfer functions $G_0(s) = \sum_{i=1}^{5} \frac{1}{s^i} H_i$ and $G(s) = C(sI - A)^{-1}B$. The plot shows an excellent agreement between the graphs for large values for s.

MATCONTROL notes: Algorithm 9.3.1 has been implemented in MATCONTROL function **minresvd**.

9.3.3 A Modified SVD Method for Minimal Realization

We describe now a modification of the above algorithm (see Juang 1994).

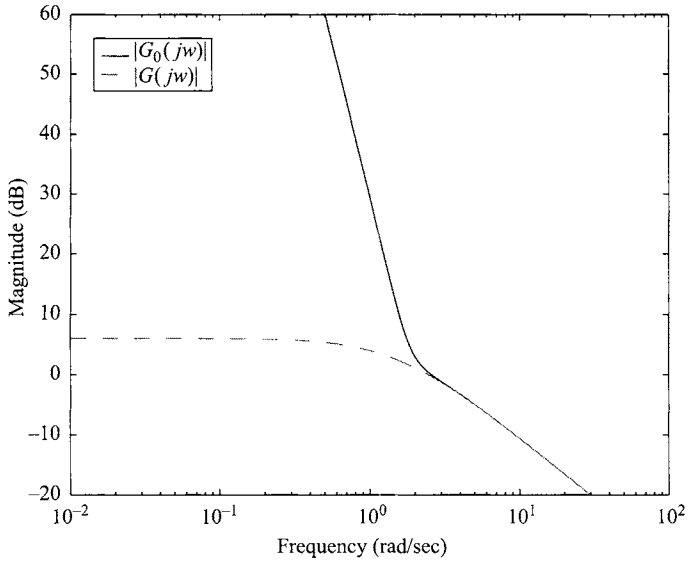

FIGURE 9.1: Comparison of transfer functions of an SVD method.

This modified algorithm uses **lower order block Hankel matrices** in computing the system matrices A, B, and C.

Define the block Hankel matrices:

$$M_R = \begin{pmatrix} H_1 & H_2 & \cdots & H_R \\ H_2 & H_3 & \cdots & H_{R+1} \\ \vdots & \vdots & & \vdots \\ H_R & H_{R+1} & \cdots & H_{2R-1} \end{pmatrix}$$

and

$$M_{R1} = \begin{pmatrix} H_2 & H_3 & \cdots & H_{R+1} \\ H_3 & H_4 & \cdots & H_{R+2} \\ \vdots & \vdots & & \vdots \\ H_{R+1} & H_{R+2} & \cdots & H_{2R} \end{pmatrix},$$

where $R \geq n$ (n is the order of the system). Denote the controllability and observability matrices by:

$$C_{\mathrm{M}}^{R} = (B, AB, \ldots, A^{R-1}B)$$

and

$$O_{\mathrm{M}}^{R} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{R-1} \end{pmatrix}.$$

Then,

$$M_{R1} = O_{\mathrm{M}}^{R}\, A\, C_{\mathrm{M}}^{R},$$

and $M_R = O_{\mathrm{M}}^{R} C_{\mathrm{M}}^{R}$.

Consider now the SVD of M_R:

$$M_R = U \Sigma V^{\mathrm{T}} = U \Sigma^{1/2} \Sigma^{1/2} V^{\mathrm{T}}.$$

This means that O_{M}^{R} is related to U and C_{M}^{R} is related to V.

Define now Σ_n by:

$$\Sigma = \begin{pmatrix} \Sigma_n & 0 \\ 0 & 0 \end{pmatrix},$$

and U_n and V_n as the matrices formed by the first n columns of U and V, respectively. Also, let the matrices E_r' and E_m' be defined as:

$$E_r'^{\mathrm{T}} = (I_r, 0, \ldots, 0), \qquad E_m'^{\mathrm{T}} = (I_m, 0, \ldots, 0),$$

where I_s stands for identity matrix of order s, and m and r denote, respectively, the number of inputs and the number of outputs.

Then one can choose $O_{\mathrm{M}}^{R} = U_n \Sigma_n^{1/2}$ and $C_{\mathrm{M}}^{R} = \Sigma_n^{1/2} V_n^{\mathrm{T}}$.

From the equation:

$$M_R = O_{\mathrm{M}}^{R} C_{\mathrm{M}}^{R} = U_n \Sigma_n^{1/2} \Sigma_n^{1/2} V_n^{\mathrm{T}}$$

it follows that B and C can be chosen as:

$$B = \Sigma_n^{1/2} V_n^{\mathrm{T}} E_m' \quad \text{and} \quad C = E_r'^{\mathrm{T}} U_n \Sigma_n^{1/2}.$$

Also, from the equation:

$$M_{R1} = O_{\mathrm{M}}^{R} A C_{\mathrm{M}}^{R} = U_n \Sigma_n^{1/2} A \Sigma_n^{1/2} V_n^{\mathrm{T}}$$

it follows that

$$A = \Sigma_n^{-1/2} U_n^{\mathrm{T}} M_{R1} V_n \Sigma_n^{-1/2}.$$

Thus, we have the following modified algorithm using the SVD of lower order Hankel matrices.

Algorithm 9.3.2. *A Modified SVD Algorithm for Minimal Realization*

Inputs. *The Markov parameters* $\{H_1, H_2, \ldots, H_{2R}\}$, $R \geq n$ *(the order of the system to be identified).*

Outputs. A, B, C *of a Minimal Realization.*

Step 1. *Form the Hankel matrices* M_R *and* M_{R1} *as defined above.*

Step 2. *Find the SVD of* M_R*:*

$$M_R = U \begin{pmatrix} \Sigma_n & 0 \\ 0 & 0 \end{pmatrix} V^{\mathrm{T}},$$

where $\Sigma_n = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$; $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$.

Step 3. *Compute*

$$\begin{aligned} A &= \Sigma_n^{-1/2} U_n^{\mathrm{T}} M_{R1} V_n \Sigma_n^{-1/2}, \\ B &= \Sigma_n^{1/2} V_n^{\mathrm{T}} E'_m, \\ C &= E_r'^{\mathrm{T}} U_n \Sigma_n^{1/2}, \end{aligned}$$

where U_n *and* V_n *are the matrices of the first n columns of* U *and* V*, respectively, and* E'_m *and* E'_r *are as defined above.*

Example 9.3.2. We consider the previous example again. Take $R = n = 2$. Let $m = 1, r = 1$.

Then,

$$M_R = \begin{pmatrix} H_1 & H_2 \\ H_2 & H_3 \end{pmatrix} = \begin{pmatrix} 3 & 5 \\ 5 & 9 \end{pmatrix},$$

$$M_{R1} = \begin{pmatrix} H_2 & H_3 \\ H_3 & H_4 \end{pmatrix} = \begin{pmatrix} 5 & 9 \\ 9 & 17 \end{pmatrix}.$$

$$\Sigma_2 = \mathrm{diag} \begin{pmatrix} 11.8310 & 0 \\ 0 & 0.1690 \end{pmatrix}$$

$$U_2 = \begin{pmatrix} -0.4927 & -0.8702 \\ -0.8702 & 0.4927 \end{pmatrix}$$

$$V_2 = \begin{pmatrix} -0.4927 & -0.8702 \\ -0.8702 & 0.4927 \end{pmatrix}.$$

$$A = \Sigma_2^{-1/2} U_2^{\mathrm{T}} M_{R1} V_2 \Sigma_2^{-1/2} = \begin{pmatrix} 1.8430 & -0.3638 \\ -0.3638 & 1.1570 \end{pmatrix},$$

$$B = \Sigma_2^{1/2} V_2^{\mathrm{T}} E'_1 = \begin{pmatrix} -1.6947 \\ -0.3578 \end{pmatrix},$$

$$C = E_1'^{\mathrm{T}} U_2 \Sigma_2^{1/2} = (-1.6947 \ \ -0.3578).$$

Verification:

$$E_1 = CB - H_1 = O(10^{-15})$$
$$E_2 = CAB - H_2 = O(10^{-15})$$
$$E_3 = CA^2B - H_3 = O(10^{-15})$$
$$E_4 = CA^3B - H_4 = O(10^{-14})$$
$$E_5 = CA^4B - H_5 = O(10^{-14}).$$

Remarks

- Algorithm 9.3.2, when extended to reconstruct the Markov parameters of a reduced-order system obtained by eliminating "noisy modes," is called **Eigensystem Realization Algorithm** (ERA) because information from the eigensystem of the realized state matrix obtained in Algorithm 9.3.2 is actually used to obtain the reduced-order model. The details can be found in Juang (1994, pp. 133–144).
- The optimal choice of the number R requires engineering intuition. The choice has to be made based on measurement data to minimize the size of the Hankel matrix M_R. See Juang (1994).

Figure 9.2 shows a comparison between the graphs of the transfer function $G_0(s) = \sum_{i=1}^{4} \frac{1}{s^i} H_i$ and $G(s) = C(sI - A)^{-1}B$. The plot shows an excellent agreement between the graphs for large values of s.

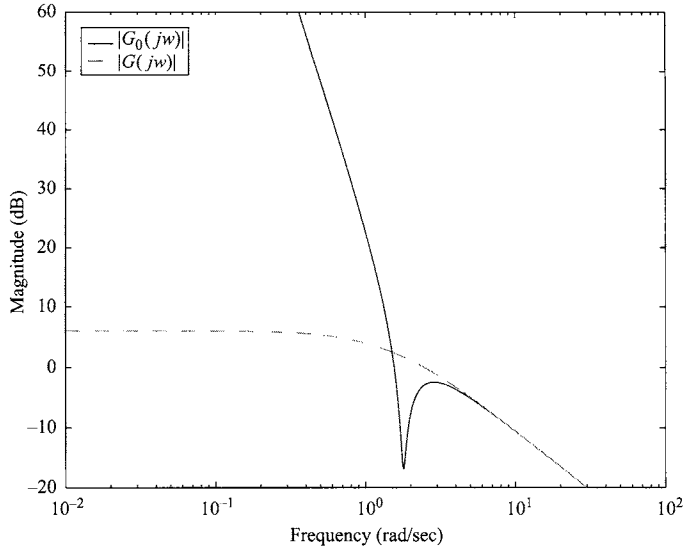

FIGURE 9.2: Comparison of transfer functions of a modified SVD method.

MATCONTROL note: Algorithm 9.3.2 has been implemented in MATCONTROL function **minremsvd**.

9.4 SUBSPACE IDENTIFICATION ALGORITHMS

In this section we consider the problem of identifying the system matrices of an unknown system, **given a large number of input and output measurements. The problem is important in practical applications because it avoids computations of Markov parameters.**

We state two SVD-based subspace algorithms, one for the deterministic case and another for the stochastic case. First, we consider the deterministic case.

Specifically, the deterministic identification problem is:

> Given a large number of input and output measurements, u_k and y_k, respectively of the unknown system:
>
> $$x_{k+1} = Ax_k + Bu_k,$$
> $$y_k = Cx_k + Du_k,$$
>
> determine the order n of the unknown system and the system matrices $\{A, B, C, D\}$ up to within a similarity transformation; $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{r\times n}$, and $D \in \mathbb{R}^{r\times m}$.

9.4.1 A Subspace Deterministic Model Identification Algorithm

The algorithm has two major steps:

First, a state vector sequence is constructed as the intersection of the row spaces of two block Hankel matrices, constructed from the input/output data.

Second, the system matrices A, B, C, and D are obtained from the least-squares solution of a set of linear equations.

There exists different ways to compute the intersection (see Van Overschee and De Moor (1996a, 1996b for details and references). One way, presented in Moonen *et al.* (1989) is via the SVD of a concatenated Hankel matrix composed of two Hankel matrices defined by the input and output data, as follows:

$$H_{k|k+i} = \begin{pmatrix} Y_{k|k+i} \\ U_{k|k+i} \end{pmatrix}, \qquad H_{k+1|k+2i} = \begin{pmatrix} Y_{k|k+2i} \\ U_{k|k+2i} \end{pmatrix},$$

where

$$Y_{k|k+i} = \begin{pmatrix} y_k & y_{k+1} & \cdots & y_{k+j-1} \\ y_{k+1} & y_{k+2} & \cdots & y_{k+j} \\ y_{k+2} & y_{k+3} & \cdots & y_{k+j+1} \\ \vdots & \vdots & & \vdots \\ y_{k+i-1} & y_{k+i} & \cdots & y_{k+j+i-2} \end{pmatrix}$$

and

$$Y_{k|k+2i} = \begin{pmatrix} y_{k+i} & y_{k+i+1} & \cdots & y_{k+i+j-1} \\ y_{k+i+1} & y_{k+i+2} & \cdots & y_{k+i+j} \\ \vdots & \vdots & \vdots & \vdots \\ y_{k+2i-1} & y_{k+2i} & \cdots & y_{k+2i+j-2} \end{pmatrix}.$$

The matrices $U_{k|k+i}$ and $U_{k|k+2i}$ are similarly defined from the input data. Let $(X = (x_k, x_{k+1}, \ldots, x_{k+j-1}))$.

The following assumptions are made:

- $\text{rank}(X) = n$ (n is the minimal system order)
- $\text{span}_{\text{row}}(X) \cap \text{span}_{\text{row}}(U_{k|k+i}) = \emptyset$
- $\text{rank}(U_{k|k+i}) =$ Number of rows in $U_{k|k+i}$.

Theorem 9.4.1. *Let the SVD of*

$$H = \begin{pmatrix} H_{k|k+i} \\ H_{k+1|k+2i} \end{pmatrix}$$

be

$$H = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix} \begin{pmatrix} S_{11} & 0 \\ 0 & 0 \end{pmatrix} V^{\mathrm{T}}.$$

Then the state vector sequence $X_2 = (x_{k+i}, x_{k+i+1}, \ldots, x_{k+i+j-1})$ is given by

$$X_2 = U_q^{\mathrm{T}} U_{12}^{\mathrm{T}} H_{k|k+i},$$

where U_q is defined by the SVD of $U_{12}^{\mathrm{T}} U_{11} S_{11}$:

$$U_{12}^{\mathrm{T}} U_{11} S_{11} = \begin{pmatrix} U_q, & U_q^{\perp} \end{pmatrix} \begin{pmatrix} S_q & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_q^{\mathrm{T}} \\ V_q^{\perp \mathrm{T}} \end{pmatrix}.$$

Proof. It can be shown (**Exercise 9.12**) that

$$\text{span}_{\text{row}}(X_2) = \text{span}_{\text{row}}(H_{k|k+i}) \cap \text{span}_{\text{row}}(H_{k+1|k+2i}).$$

Thus, X_2 can be realized as a basis for the row space of $U_{12}^{\mathrm{T}} H_{k|k+i}$. Then taking the SVD of $U_{12}^{\mathrm{T}} H_{k|k+i}$, we have

$$\begin{aligned} U_{12}^{\mathrm{T}} H_{k|k+i} &= U_{12}^{\mathrm{T}} (U_{11} \;\; U_{12}) \begin{pmatrix} S_{11} & 0 \\ 0 & 0 \end{pmatrix} V^{\mathrm{T}}, \\ &= (U_{12}^{\mathrm{T}} U_{11} S_{11} \;\; 0) V^{\mathrm{T}}, \\ &= (U_q S_q V_q^{\mathrm{T}} \;\; 0) V^{\mathrm{T}}, \\ &= U_q (S_q \;\; 0)(V V_q)^{\mathrm{T}}. \end{aligned}$$

Noting that $U_q^{\mathrm{T}} U_q = I_{n\times n}$, we obtain from above

$$U_q^{\mathrm{T}} U_{12}^{\mathrm{T}} H_{k|k+i} = (S_q \;\; 0)(VV_q)^{\mathrm{T}},$$

which is a basis for the row space of $U_{12}^{\mathrm{T}} H_{k|k+i}$ and therefore is a realization of X_2. ■

Once X_2 is determined, the system matrices A, B, C, and D are identified by solving (**in the least-squares sense**) the following overdetermined set of linear equations:

$$\begin{pmatrix} x_{k+i+1} & \cdots & x_{k+i+j-1} \\ y_{k+i} & \cdots & y_{k+i+j-2} \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} x_{k+i} & \cdots & x_{k+i+j-2} \\ u_{k+i} & \cdots & u_{k+i+j-2} \end{pmatrix}.$$

It is, however, shown in De Moor *et al.* (1999) that the state vector sequence X_2 does not need to be explicitly computed. The system matrices A, B, C, and D may be identified by making use of the already computed SVD of H. The above set of equations may then be replaced by an equivalent reduced set of equations (see **Algorithm 9.4.1**).

This way of determining A, B, C, and D is computationally more efficient.

To do this, it is useful to redefine the matrices $H_{k|k+i}$ and $H_{k+i|k+2i}$ as follows:

$$H_{k|k+i} = \begin{pmatrix} u_k & u_{k+1} & \cdots & u_{k+j-1} \\ y_k & y_{k+1} & \cdots & y_{k+j-1} \\ u_{k+1} & u_{k+2} & \cdots & u_{k+j} \\ y_{k+1} & y_{k+2} & \cdots & y_{k+j} \\ \vdots & \vdots & & \vdots \\ u_{k+i-1} & u_{k+i} & \cdots & u_{k+i+j-2} \\ y_{k+i-1} & y_{k+i} & \cdots & y_{k+i+j-2} \end{pmatrix},$$

$$H_{k+i|k+2i} = \begin{pmatrix} u_{k+i} & u_{k+i+1} & \cdots & u_{k+i+j-1} \\ y_{k+i} & y_{k+i+1} & \cdots & y_{k+i+j-1} \\ u_{k+i+1} & u_{k+i+2} & \cdots & u_{k+i+j} \\ y_{k+i+1} & y_{k+i+2} & \cdots & y_{k+i+j} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ u_{k+2i-1} & u_{k+2i} & \cdots & u_{k+2i+j-2} \\ y_{k+2i-1} & y_{k+2i} & \cdots & y_{k+2i+j-2} \end{pmatrix}.$$

The above theorem still remains valid.

The following notation will be needed to state the algorithm.

$M(p:q, l:s)$ is the submatrix of M at the intersection of rows $p, p+1, \ldots, q$ and columns $l, l+1, \ldots, s$; $M(:, l:s)$ is the submatrix of M containing

columns $l, l+1, \ldots, s$ and $M(p:q,:)$ is the submatrix of M containing rows $p, p+1, \ldots, q$.

Algorithm 9.4.1. *A Deterministic Subspace Identification Algorithm.*

Inputs. *The input and output sequence $\{u_k\}$ and $\{y_k\}$, respectively. The integers $i \geq n$, where n is the order of the system to be identified and j.*

Outputs. *The identified system matrices A, B, C, and D.*

Assumptions.

1. *The system is observable.*
2. *The integers i and j are sufficiently large, and in particular $j >> \max(mi, ri)$, where m and r are the number of inputs and outputs.*

Step 1. *Calculate U and S from the SVD of H, where*

$$H = \begin{pmatrix} H_{k|k+i} \\ H_{k+1|k+2i} \end{pmatrix}.$$

$$H = USV^{\mathrm{T}} = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix} \begin{pmatrix} S_{11} & 0 \\ 0 & 0 \end{pmatrix} V^{\mathrm{T}}.$$

(Note that the dimensions of U_{11}, U_{12}, and S_{11} are, respectively, $(mi + ri) \times (2mi + n)$, $(mi + ri) \times (2ri - n)$, and $(2mi + n) \times (2mi + n)$).

Step 2. *Calculate the SVD of $U_{12}^{\mathrm{T}} U_{11} S_{11}$:*

$$U_{12}^{\mathrm{T}} U_{11} S_{11} = (U_q, U_q^{\perp}) \begin{pmatrix} S_q & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_q^{\mathrm{T}} \\ V_q^{\perp \mathrm{T}} \end{pmatrix}.$$

Step 3. *Solve the following set of linear equations for A, B, C, and D (in the least-squares sense):*

$$\begin{pmatrix} U_q^{\mathrm{T}} U_{12}^{\mathrm{T}} U(mi + ri + 1 : (m + r)(i + 1), :)S \\ U(mi + ri + m + 1 : (m + r)(i + 1), :)S \end{pmatrix}$$
$$= \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} U_q^{\mathrm{T}} U_{12}^{\mathrm{T}} U(1 : mi + ri :)S \\ U(mi + ri + 1 : mi + ri + m, :)S \end{pmatrix}.$$

Remark

- It is to be noted that the system matrices are determined from U and S only; the larger matrix V is never used in the computations. Since the matrix H whose SVD to be computed could be very large in practice, computing U and S only, without computing the full SVD of H, will be certainly very useful in practice. **Also, as stated before, the state vector sequence X_2 is not explicitly computed.**

There exists also an on-line version of the above algorithm. See Moonen *et al.* (1989) for details.

Example 9.4.1. Consider the following input–output data

$$u = \begin{pmatrix} 0.09130 & 0.1310 & 0.6275 & 0.1301 & -0.2206 & 0.1984 & 0.4081 & -0.0175 \\ 0.2766 & 0.7047 & 0.9173 & 0.9564 & 0.6631 & 0.7419 & 0.7479 & 1.2133 \\ 1.2427 & 1.2942 & 1.3092 & 1.1574 & 1.5600 & 1.0913 & 0.7765 & \end{pmatrix}^{\mathrm{T}},$$

$$y = \begin{pmatrix} 0.6197 & -0.4824 & 0.3221 & 0.2874 & -0.4582 & -0.1729 & 0.3162 & 0.0946 \\ -0.3497 & 0.3925 & 0.2446 & 0.2815 & 0.05621 & -0.2201 & 0.1397 & -0.0880 \\ 0.5250 & -0.1021 & 0.2294 & -0.0616 & -0.0706 & 0.3982 & -0.5695 & \end{pmatrix}^{\mathrm{T}},$$

generated from the discrete-time system:

$$x_{k+1} = \begin{pmatrix} -0.2 & 0.3 \\ 1 & 0 \end{pmatrix} x_k + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u_k,$$

$$y_k = (1, \ -1)\, x_k.$$

Step 1. $S = \mathrm{diag}(9.1719, 1.9793, 1.8031, 1.6608, 1.4509, 1.3426, 1.2796, 1.0657, 0.5012, 0.4554, 5.1287 \times 10^{-16}, 3.5667 \times 10^{-16}, 2.2847 \times 10^{-16}, 1.3846 \times 10^{-16}, 9.8100 \times 10^{-17}, 1.0412 \times 10^{-18})$.

Step 2.

$$U_{12} = \begin{pmatrix} 0.4392 & -0.0372 & -0.1039 & 0.2139 & -0.0297 & -0.3324 \\ -0.1318 & 0.01116 & 0.0312 & -0.0642 & 0.0090 & 0.0997 \\ -0.3277 & 0.3783 & 0.1880 & 0.2071 & 0.3299 & 0.2790 \\ 0.0544 & -0.1098 & -0.0460 & -0.0836 & -0.0960 & -0.0505 \\ -0.2282 & -0.4853 & -0.2806 & 0.1674 & -0.3377 & 0.1086 \\ 0.4965 & 0.0743 & -0.0282 & 0.1215 & 0.0416 & -0.3597 \\ 0.4062 & -0.0910 & 0.3218 & -0.3029 & 0.1580 & 0.3608 \\ 0.0012 & 0.3828 & 0.0071 & 0.4531 & 0.2565 & -0.1672 \end{pmatrix},$$

$$U_{11} = \begin{pmatrix} 0.2417 & -0.2139 & 0.2202 & 0.3049 & -0.4614 & -0.1558 & 0.0638 & 0.0142 & -0.2980 & -0.2488 \\ 0.0211 & 0.3581 & 0.1578 & 0.3546 & -0.4804 & -0.0503 & -0.1130 & 0.2833 & 0.1475 & 0.5831 \\ 0.2768 & -0.3634 & -0.2593 & -0.0824 & -0.3694 & 0.2078 & 0.0116 & -0.0870 & -0.1208 & 0.0008 \\ 0.0383 & -0.5002 & 0.1157 & -0.0384 & -0.1351 & -0.0780 & 0.2107 & -0.2894 & 0.7057 & 0.1893 \\ 0.3109 & 0.0687 & -0.5137 & 0.1861 & -0.0370 & 0.0372 & 0.2759 & -0.0140 & -0.0267 & -0.0101 \\ 0.0338 & 0.0580 & -0.4635 & -0.2733 & -0.0251 & 0.3664 & -0.1282 & 0.0417 & 0.0803 & 0.3867 \\ 0.3402 & 0.0787 & -0.0426 & 0.4406 & 0.2193 & 0.2340 & -0.0493 & -0.2103 & 0.0708 & -0.0626 \\ 0.0389 & 0.2704 & -0.1147 & 0.3117 & 0.2969 & -0.2668 & 0.3531 & -0.0220 & 0.2897 & -0.0315 \end{pmatrix},$$

$$S_q = \mathrm{diag}(1.94468, \ 0.624567).$$

Step 3.

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \left(\begin{array}{cc|c} 0.2635 & 0.1752 & -0.4644 \\ 1.0153 & -0.4635 & 0.5503 \\ \hline -0.1780 & 1.6527 & 1.4416 \times 10^{-16} \end{array}\right).$$

Verification: The first 10 Markov parameters (denoted by $H_i, i = 1, \ldots, 10$) of the original system and those of the identified system (denoted by $H_i', i = 1, \ldots, 10$) are given below:

$$\begin{aligned} H_1 &= 1, \ H_1' = 0.9922, \\ H_2 &= -1.2, \ H_2' = -1.1962, \\ H_3 &= 0.5400, \ H_3' = 0.5369, \\ &\vdots \\ H_{10} &= -0.0343, \ H_{10}' = -0.0341. \end{aligned}$$

9.4.2 A Stochastic Subspace Model Identification Algorithm

We now consider the stochastic case:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k, \\ y_k &= Cx_k + Du_k + v_k, \end{aligned}$$

where $v_k \in \mathbb{R}^{r\times 1}$ and $w_k \in \mathbb{R}^{n\times 1}$ are unobserved vector signals; v_k is the measurement noise and w_k is the process noise. It is assumed that

$$E\left[\begin{pmatrix} w_p \\ v_p \end{pmatrix}\begin{pmatrix} w_q^{\mathrm{T}} & v_q^{\mathrm{T}} \end{pmatrix}\right] = \begin{pmatrix} Q & S \\ S^{\mathrm{T}} & R \end{pmatrix}\delta_{pq} \geq 0,$$

where the matrices Q, R, and S are covariance matrices of the noise sequences w_k and v_k. The problem is now to determine the system matrices A, B, C, and D up to within a similarity transformation and also the covariance matrices Q, S, and R, given a large number of input and output data u_k and y_k, respectively.

We state a subspace algorithm for the above problem taken from the recent paper of DeMoor *et al.* (1999). The algorithm, as in the deterministic case, determines the system matrices by first finding an estimate $\tilde{X}_f$ of the state sequence $\tilde{X}$ from the measurement data.

The sequence $\tilde{X}_f$ is determined using certain oblique projections.

Define the input Hankel matrix $U_{i|l}$ from the input data as:

$$U_{k|l} = \begin{pmatrix} u_k & u_{k+1} & \cdots & u_{k+j-1} \\ u_{k+1} & u_{k+2} & \cdots & u_{k+j} \\ \vdots & & & \\ u_l & u_{l+1} & \cdots & u_{l+j-1} \end{pmatrix}.$$

Similarly, define the output Hankel matrix $Y_{k|l}$ from the output data.

The matrices A, B, C, D are then determined by solving the least-squares problem:

$$\min_{A,B,C,D} \| \begin{pmatrix} \tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} - \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} \tilde{X}_i \\ U_{i|i} \end{pmatrix} \|_F^2 .$$

Once the system matrices are obtained by solving above least-squares problem, the noise covariances Q, S, and R can be estimated from the residuals (**see Algorithm 9.4.2 below for details**).

The algorithm, in particular, can be used to solve the deterministic problem. Thus, it can be called a **combined deterministic stochastic identification algorithm**.

Definition 9.4.1. *The **oblique projection** of $A \in \mathbb{R}^{p \times j}$ along the row space of $B \in \mathbb{R}^{q \times j}$ on the row space of $C \in \mathbb{R}^{r \times j}$, denoted by $A/_B C$ is defined as:*

$$A/_B \, C = A(C^T \;\; B^T) \left[\begin{pmatrix} CC^T & CB^T \\ BC^T & BB^T \end{pmatrix}^\dagger \right]_{first \;\; r \; columns} C,$$

where † denotes the Moore–Penrose pseudo-inverse of the matrix.

For convenience, following the notations of the above paper, we write

$$U_p = U_{0|i-1}, \qquad U_f = U_{i|2i-1},$$
$$Y_p = Y_{0|i-1}, \qquad Y_f = Y_{i|2i-1},$$

where the subscript p and f denote, respectively, the past and the future. The matrix containing the past inputs U_p and outputs Y_p will be called W_p:

$$W_p = \begin{pmatrix} Y_p \\ U_p \end{pmatrix}.$$

The matrices $W_{0|i-1}$ and $W_{0|i}$ are defined in the same way as $U_{0|i-1}$ and $U_{0|i}$ from Y_p and W_p.

The following assumptions are made:

- The input u_k is uncorrelated with the noise w_k and v_k.
- The input covariance matrix $(1/j)(U_{0|2i-1}U_{0|2i-1}^T)$ is of full rank, that is, the rank is $2mi$ (the sequence u_k is then called **persistently exciting** of order $2i$).
- The number of available measurements is sufficiently large, so that $j \to \infty$.
- The noise w_k and v_k are not identically zero.

Algorithm 9.4.2. *A Subspace Stochastic Identification Algorithm.*
Inputs. *The input and output sequences $\{u_k\}$ and $\{y_k\}$.*
Outputs. *The order of the system and the system matrices A, B, C, D.*
Assumptions. *As above.*

Step 1. *Find the oblique projections:*

$$\mathcal{O}_i = Y_{i|2i-1}/_{U_{i|2i-1}} \boldsymbol{W}_{0|i-1}, \qquad \mathcal{O}_{i+1} = Y_{i+1|2i-1}/_{U_{i+1|2i-1}} \boldsymbol{W}_{0|i}.$$

Step 2. *Compute the SVD of the oblique projection:*

$$\mathcal{O}_i = USV^{\mathrm{T}} = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^{\mathrm{T}} \\ V_2^{\mathrm{T}} \end{pmatrix},$$

(The order n of the system is equal to the order of S_1).

Step 3. *Define Γ_i and Γ_{i-1} as:*

$$\Gamma_i = U_1 S_1^{1/2}, \qquad \Gamma_{i-1} = \underline{\Gamma_i},$$

where $\underline{\Gamma_i}$ is Γ_i without the last block row.

Step 4. *Determine the state sequences:*

$$\tilde{X}_i = S_1 V_1^{\mathrm{T}}, \qquad \tilde{X}_{i+1} = \Gamma_{i-1}^{\dagger} \mathcal{O}_{i+1}.$$

Step 5. *Solve the following linear equations* ***(in the least-squares sense)*** *for A, B, C, D and the residuals ρ_w and ρ_v:*

$$\begin{pmatrix} \tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} \tilde{X}_i \\ U_{i|i} \end{pmatrix} + \begin{pmatrix} \rho_w \\ \rho_v \end{pmatrix}.$$

Step 6. *Determine the noise covariances Q, S, and R from the residuals as:*

$$\begin{pmatrix} Q & S \\ S^{\mathrm{T}} & R \end{pmatrix}_i = \frac{1}{j} \left[\begin{pmatrix} \rho_w \\ \rho_v \end{pmatrix} \begin{pmatrix} \rho_w^{\mathrm{T}} & \rho_v^{\mathrm{T}} \end{pmatrix} \right].$$

where the index i denotes a "bias" induced for finite i, which vanishes as $i \to \infty$.

Implementational remarks: In practical implementation, Step 4 should not be computed as above, because explicit computation of the latter matrix V is time consuming.

In fact, in a good software, the oblique projections in Step 1 are computed using a fast structure preserving QR factorization method and SVD in Step 2 is applied to only a part of the R-factor from the QR factorization.

For details of the proofs and practical implementations of these and other related subspace algorithms for system identification and an account of the extensive up-to-date literature (including the software on identification) on the subject, the readers are referred to the book by Van Overschee and DeMoor (1996a) and the recent review paper by DeMoor *et al.* (1999).

MATLAB note: M-files implementing Algorithm 9.4.2 (and others) come with the book by Van Overschee and DeMoor (1996b) and can also be obtained from **ftp://www.esat.kuleuven.ac.be/pub/SISTA/vanoverschee/book/subfun/**

9.4.3 Continuous-Time System Identification

Subspace system identification algorithms, analogous to Algorithms 9.4.1 and 9.4.2, can also be developed for a continuous-time system:

$$\dot{x}(t) = Ax(t) + Bu(t),$$
$$y(t) = Cx(t) + Du(t).$$

However, the input and output matrices have to be defined differently and they need computations of derivatives. Thus, define

$$U^{\mathrm{c}}_{0|i-1} = \begin{pmatrix} u(t_0) & u(t_1) & \cdots & u(t_{j-1}) \\ u^{(1)}(t_0) & u^{(1)}(t_1) & \cdots & u^{(1)}(t_{j-1}) \\ \cdots & \cdots & \cdots & \cdots \\ u^{(i-1)}(t_0) & u^{(i-1)}(t_1) & \cdots & u^{(i-1)}(t_{j-1}) \end{pmatrix},$$

where $u^{(p)}(t)$ denotes the pth derivative of $u(t)$, and "c" stands for "**continuous**."

The matrices $Y^{\mathrm{c}}_{0|i-1}$, $U^{\mathrm{c}}_{i|2i-1}$, and X^{c}_i are similarly defined.

The continuous-time system identification problem can be stated as follows:

Given input and output measurements $u(t)$, $y(t)$, $t = t_0, t_1, \ldots, t_{j-1}$ and the estimates of the derivatives $u^{(p)}(t)$ and $y^{(p)}(t)$ up to order $2i - 1$, of the above unknown system, find the system matrices A, B, C, D, of the above continuous-time system up to within a similarity transformation.

9.4.4 Frequency-Domain Identification

The problem we consider here is the one of identifying a **continuous-time model** given a set of frequency responses. The problem can also be solved for a discrete-time system. For frequency-domain identification of discrete-time systems, see McKelvey (1994a, 1994b, 1994c). Specifically, the frequency-domain identification problem for a continuous-time system is stated as follows:

> Given N frequency domain frequency responses $G(j\omega_k)$, measured at frequencies ω_k (not necessarily distinct), $k = 1, 2, \ldots, N$, find the system matrices A, B, C, and D.

One indirect approach for solving the problem is to estimate the Markov parameters via matrix-fraction descriptions of the frequency responses $G(j\omega_k)$ and then apply any of the Markov parameters based time-domain algorithms described in Section 9.3. (See **Exercise 9.10**).

We will, however, not discuss this here. For details, the readers are referred to the book by Juang (1994). Rather, we state here a direct **subspace identification algorithm** from the paper of DeMoor *et al.* (1999).

Let $\alpha > n$ be a user supplied index. Let $\mathrm{Re}(M)$ and $\mathrm{Im}(M)$ denote, respectively, the real and imaginary parts of a complex matrix M. Define the following matrices

from the given frequency responses:

$$\mathcal{H} = \left(\mathrm{Re}(\mathcal{H}^{\mathrm{c}}),\ \mathrm{Im}(\mathcal{H}^{\mathrm{c}})\right),$$
$$\mathcal{I} = (\mathrm{Re}(\mathcal{I}^{\mathrm{c}}),\ \mathrm{Im}(\mathcal{I}^{\mathrm{c}})),$$

where

$$\mathcal{H}^{\mathrm{c}} = \begin{pmatrix} G(j\omega_1) & G(j\omega_2) & \cdots & G(j\omega_N) \\ (j\omega_1)G(j\omega_1) & (j\omega_2)G(j\omega_2) & \cdots & (j\omega_N)G(j\omega_N) \\ \vdots & \vdots & & \vdots \\ (j\omega_1)^{\alpha-1}G(j\omega_1) & (j\omega_2)^{\alpha-1}G(j\omega_2) & \cdots & (j\omega_N)^{\alpha-1}G(j\omega_N) \end{pmatrix}$$

and

$$\mathcal{I}^{\mathrm{c}} = \begin{pmatrix} I_m & I_m & \cdots & I_m \\ (j\omega_1)I_m & (j\omega_2)I_m & \cdots & (j\omega_N)I_m \\ \vdots & \vdots & & \vdots \\ (j\omega_1)^{\alpha-1}I_m & (j\omega_2)^{\alpha-1}I_m & \cdots & (j\omega_N)^{\alpha-1}I_m \end{pmatrix}.$$

Algorithm 9.4.3. *Continuous-Time Frequency-Domain Subspace Identification Algorithm.*

Inputs. *The set of measured frequencies* $G(j\omega_1), G(j\omega_2), \ldots, G(j\omega_N)$, *an integer* α *and a weighting matrix* W.

Outputs. *The system matrices* A, B, C, *and* D.

Step 1. *Find the orthogonal projection of the row space of* $\mathcal{H}$ *into the row space of* $\mathcal{I}^{\perp}$:

$$O_\alpha = \mathcal{H} - \mathcal{H}\mathcal{I}^{\dagger}\mathcal{I}.$$

Step 2. *Compute the SVD of* $W O_\alpha$:

$$W O_\alpha = USV^{\mathrm{T}} = (U_1, U_2)\begin{pmatrix} S_1 & 0 \\ 0 & S_0 \end{pmatrix}\begin{pmatrix} V_1^{\mathrm{T}} \\ V_2^{\mathrm{T}} \end{pmatrix},$$

where W *is a weighting matrix.*

Step 3. *Determine* $\Gamma_\alpha = W^{-1}U_1 S_1^{1/2}$.

Step 4. *Determine* A *and* C *as follows:*

$C =$ *the first* r *rows of* Γ_α

$A = \underline{\Gamma_\alpha}^{\dagger}\bar{\Gamma}_\alpha$,

where $\bar{\Gamma}_\alpha$ *and* $\underline{\Gamma_\alpha}$ *denote* Γ_α *without the first and last* r *rows.*

Step 5. *Determine B and D via the least-squares solution of the linear systems of equations:*

$$\begin{pmatrix} \mathrm{Re}(L) \\ \mathrm{Im}(L) \end{pmatrix} = \begin{pmatrix} \mathrm{Re}(M) \\ \mathrm{Im}(M) \end{pmatrix} \begin{pmatrix} B \\ D \end{pmatrix},$$

where L and M are given by:

$$L = \begin{pmatrix} G(j\omega_1) \\ \vdots \\ G(j\omega_N) \end{pmatrix} \quad \text{and} \quad M = \begin{pmatrix} C(j\omega_1 - A)^{-1} & I_r \\ \vdots & \\ C(j\omega_N - A)^{-1} & I_r \end{pmatrix}.$$

(*Note that L and M are, respectively, of order* $rN \times m$ *and* $rN \times (n + r)$.)

Remarks

- The choice of the weighting matrix W is very important. If W is chosen appropriately, then the results are "unbiased"; otherwise, they will be "biased." For details of how the weighting should be chosen, the readers are referred to the paper by Van Overschee and De Moor (1996a).
- The algorithm works well when n and i are small.

  However, when i grows larger, the block Hankel matrices $\mathcal{H}$ and $\mathcal{I}$ became very highly ill-conditioned. The paper of Van Overschee and De Moore (1996a) contains a more numerically effective algorithm.

9.5 SOME SELECTED SOFTWARE

9.5.1 MATLAB Control System Toolbox

State-space models

minreal—Minimal realization and pole/zero cancellation
augstate—Augment output by appending states.

9.5.2 MATCONTROL

MINRESVD—Finding minimal realization using SVD of Hankel matrix of Markov parameters (**Algorithm 9.3.1**)
MINREMSVD—Finding minimal realization using SVD of Hankel matrix of lower order (**Algorithm 9.3.2**).

9.5.3 CSP-ANM

Model identification

- The system identification from its impulse responses is performed by `ImpulseResponseIdentify` [*response*].

- The system identification from its frequency responses is performed by `FrequencyResponseIdentify` [*response*].
- The system identification directly from input–output data is performed by `OutputResponseIdentify` [u, y].

9.5.4 SLICOT

Identification
IB—Subspace Identification

Time invariant state-space systems

IB01AD—Input–output data preprocessing and finding the system order
IB01BD—Estimating the system matrices, covariances, and Kalman gain
IB01CD—Estimating the initial state and the system matrices B and D.

TF—Time response

TF01QD Markov parameters of a system from transfer function matrix
TF01RD Markov parameters of a system from state-space representation

In addition to the above-mentioned software, the following toolboxes, especially designed for system identification are available.

- **MATLAB System Identification Toolbox**, developed by Prof. Lennart Ljung. (Website: http://www.mathworks.com)
- **ADAPTX**, developed by W.E. Larimore. (Website: http://adaptics.com)
- **Xmath Interactive System Identification Module**, described in the manual X-Math Interactive System Identification Module, Part 2, by P. VanOverschee, B. DeMoor, H. Aling, R. Kosut, and S. Boyd, Integrated Systems Inc., Santa Clara, California, USA, 1994 (website: http://www.isi.com/products/MATRIX$_X$/Techspec/MATRIX$_X$-Xmath/xm36.html, -/MATRIX$_X$ XMATH/inline images/pg. 37 img.html and-/MATRIX$_X$-XMath/inlineimages/pg. 38img.html).

For more details on these software packages, see the paper by DeMoor *et al.* (1999).

9.5.5 MATRIX$_X$

Purpose: Compute the minimal realization of a system.

Syntax: [SMIN, NSMIN, T]=MINIMAL (S, NS, TOL) or
[NUMMIN, DENMIN]=MINIMAL (NUM, DEN, TOL)

9.6 SUMMARY AND REVIEW

This chapter is concerned with state-space realization and model identification.

Realization

Given a transfer function matrix $G(s)$, the realization problem is the problem of finding the system matrices A, B, C, and D such that $G(s) = C(sI - A)^{-1}B + D$.

For a given proper rational function $G(s)$, there always exists a state-space realization. However, **such a realization is not unique.** In **Section 9.2.1**, the nonuniqueness of a realization is demonstrated by computing the two realizations of the same transfer function matrix $G(s)$: **controllable and observable realizations.**

Minimal Realization

A realization (A, B, C, D) of $G(s)$ is an **MR** if A has the smallest possible dimension. An important result on MR is that a **realization is minimal if and only if (A, B) is controllable and (A, C)** is **observable (Theorem 9.2.1)**.

Two MRs are related by a nonsingular transforming matrix T **(Theorem 9.2.2)**.

There are many methods for computing an MR, given a set of **Markov parameters** $H_k = CA^{k-1}B, k = 1, 2, 3, \ldots$, assuming that these Markov parameters are easily obtainable from a given transfer function. Most of these methods find an MR by factoring the Hankel matrix of Markov parameters:

$$M_k = \begin{pmatrix} H_1 & H_2 & \cdots & H_k \\ H_2 & H_3 & \cdots & H_{k+1} \\ \vdots & & & \\ H_k & H_{k+1} & \cdots & H_{2k-1} \end{pmatrix}.$$

Some basic properties of this Hankel matrix M_k that play an important role in the development of these algorithms are stated and proved in **Theorem 9.3.1**.

Two numerically viable SVD-based methods for computing an MR are given in Sections 9.3.2 and 9.3.3 **(Algorithms 9.3.1 and 9.3.2)**.

Time-Domain Subspace Identification

Many times, the Markov parameters are not easily accessible. In these cases, the system matrices must be identified from a given set of input and output data.

Two subspace algorithms for system identification: **Algorithm 9.4.1** for *deterministic identification* and **Algorithm 9.4.2** for *combined deterministic and stochastic identification* are described in **Section 9.4.**

It is assumed that the number of input and output data are very large (goes to infinity) and that the data are ergodic.

Each of these two subspace algorithms comes in two steps. The first step consists of finding (implicitly or explicitly) some estimate $\tilde{X}_i$ of the state sequence, while

in the second step, the system matrices A, B, C, and D are obtained by solving an overdetermined system (in the least-squares sense) using this state sequence $\tilde{X}_i$.

Frequency-Domain Subspace Identification

Finally, frequency-domain subspace identification is considered in **Section 9.4.4**. The problem considered there is:

Given N frequency domain responses $G(j\omega_k)$, measured at frequencies ω_k, $k = 1, 2, \ldots, N$; find the system matrices A, B, C, and D.

A **continuous-time** frequency-domain subspace identification algorithm (**Algorithm 9.4.3**) is described in **Section 9.4.4**.

9.7 CHAPTER NOTES AND FURTHER READING

Realization theory is a classical topic in system identification. Ho and Kalman (1966) first introduced the important principles and concepts of MR theory. There are now well-known books and papers in this area such as Kung (1978), Ljung (1987, 1991a, 1991b), Silverman (1971), Zeiger and McEwen (1974), Dickinson *et al.* (1974a, 1974b), Juang (1994), Norton (1986), Aström and Eykhoff (1971), Eykhoff (1974), Rissanen (1971), DeJong (1978), Brockett (1978), Datta (1980), Gragg and Lindquist (1983). These papers and books provide a good insight into the subject of system identification from Markov parameters. The paper by Gragg and Lindquist (1983) deals with partial realization problem. The subspace system identification algorithms are the input-state-output generalizations of the realization theory and these algorithms are relatively modern.

Material on subspace algorithms in this book has been taken mostly from the recent book by Van Overschee and De Moor (1996b) and the recent review paper by De Moor *et al.* (1999). **Both references contain an up-to-date extensive list of papers and books on realization theory** and **subspace identification algorithms** (see also the papers by Lindquist and Picci (1993, 1994)). Frequency-domain identification is dealt with in some depth in the book by Juang (1994) and a Newton-type algorithm for fitting transfer functions to frequency-response measurements appears in Spanos and Mingori (1993).

There exists an intimate relation between subspace system identification and frequency weighted model reduction. The frequency weighted model reduction is discussed in Chaper 14 of this book. For details of the connection between these topics, see **Chapter 5** of the book by Van Overschee and De Moor (1996b).

Exercises

9.1 Prove that there always exists a state-space realization for a proper rational function.

9.2 Verify that the controllable realization (A, B, C, D) and the observable realization (A', B', C', D') described in Section 9.2.1 are state-space realizations of the same transfer matrix $G(s)$.

9.3 Give a complete proof of Theorem 9.2.2.

9.4 Let $G(s)$ be the transfer matrix of a SISO system and let (A, b, c, d) be a state-space realization of $G(s)$:

$$G(s) = d + c(sI - A)^{-1}b = d + \frac{b(s)}{a(s)}.$$

Prove that the realization is minimal if and only if $a(s)$ and $b(s)$ are coprime.

9.5 *Generating the Markov Parameters*

(a) Show that for the discrete-time system (9.3.1) with initial condition $x_0 = 0$, the Markov parameters $H_0 = D$, $H_i = CA^{i-1}B$, $i = 1, 2, \ldots, l-1$ can be determined by solving the system:

$$y = SU,$$

where $y = (y_0, y_1, y_2, \ldots, y_{l-1})_{r \times l}$

$$S = (H_0, H_1, H_2, \ldots, H_{l-1}),$$

$$U = \begin{pmatrix} u_0 & u_1 & u_2 & \cdots & u_{l-1} \\ & u_0 & u_1 & \cdots & u_{l-2} \\ & & \ddots & & \vdots \\ & & & \ddots & \vdots \\ & & & & u_0 \end{pmatrix}_{ml \times l},$$

where m is the number of inputs and r is the number of outputs; the matrix U is an $ml \times l$ block upper triangular matrix.

(b) Assume that $A^k \approx 0$ for all time steps $k \geq p$, that is, A is discrete stable, then show that the above system can be reduced to

$$y = S'U',$$

where $y = (y_0, y_1, \ldots, y_{l-1})$,

$$S' = (H_0, H_1, H_2, \ldots, H_p)$$

and

$$U' = \begin{pmatrix} u_0 & u_1 & u_2 & \cdots & u_p & \cdots & u_{l-1} \\ & u_0 & u_1 & \cdots & u_{p-1} & \cdots & u_{l-2} \\ & & \ddots & & & & \vdots \\ & & & \ddots & & & \vdots \\ & & & & \ddots & & \vdots \\ 0 & & & & & \cdot & u_{l-p-1} \end{pmatrix}.$$

(Note that U' is of order $m(p+1) \times l$ and S' is of order $r \times m(p+1)$.)

(c) Discuss the numerical difficulties in solving the above system and work out an example to illustrate the difficulties.

9.6 Prove that the Hankel matrix M_k can be decomposed in the form (9.3.3).

9.7 Assuming that $H_k \rightarrow 0$ as $k \rightarrow \infty$, prove that the realization obtained by Algorithm 9.3.1 is discrete-stable. (**Hint:** Show that $\| S^{-1/2} A S^{1/2} \|_2$ is less than unity.)

9.8 (a) Construct a discrete-time system:

$$x_{k+1} = Ax_k + Bu_k$$
$$y_k = Cx_k$$

with suitable randomly generated matrices A, B, and C.

(b) Construct sufficient number of Markov parameters using the inputs $u_0 = 1, u_i = 0, i > 1$, and assuming zero initial condition.

(c) Apply Algorithms 9.3.1 and 9.3.2 to identify the system matrices A, B, and C.

(d) In each case, plot the transfer function of the original and the identified model.

9.9 A stable system is **balanced** if both controllability and observability Grammians are equal to a diagonal matrix **(Chapter 14)**.
Prove that if Algorithm 9.3.2 starts with the Hankel matrix:

$$M_{\beta,\alpha} = \begin{pmatrix} H_1 & H_2 & \cdots & H_\beta \\ H_2 & H_3 & \cdots & H_{\beta+1} \\ \vdots & & & \\ H_\alpha & H_{\alpha+1} & \cdots & H_{\alpha+\beta-1} \end{pmatrix},$$

then the algorithm gives a balanced realization when the indices α and β are sufficiently large.

9.10 Frequency-Domain Realization using Markov Parameters (Juang (1994)).
Consider the frequency response function $G(z_k) = C(zI - A)^{-1}B + D$; $z_k = e^{j2\pi k/l}$, where l is the data length and $z_k, k = 0, 1, \ldots, l$ correspond to the frequency points at $2\pi k/l\Delta t$, with Δt being the sampling time interval.
Write $G(z_k) = Q^{-1}(z_k)R(z_k)$
where

$$Q(z_k) = I_r + Q_1 z_k^{-1} + \cdots + Q_p z_k^{-p}$$
$$R(z_k) = R_0 + R_1 z_k^{-1} + \cdots + R_p z_k^{-p}$$

are matrix polynomials and I_r is the identity matrix of order r.

(a) Prove that knowing $G(z_k)$, the coefficient matrices of $Q(z_k)$ and $R(z_k)$ can be found by solving a least-squares problem.

(b) How can the complex arithmetic be avoided in part (a)?

(c) Show how to obtain the Markov parameters from the coefficient matrices found in (a).
(**Hint:** $(\Sigma_{i=0}^{p} Q_i z^{-i})(\Sigma_{i=0}^{\infty} H_i z^{-i}) = \Sigma_{i=0}^{p} R_i z^{-i}$).

(d) Derive an algorithm for frequency-domain realization similar to Algorithm 9.3.2 based on (a)–(c).

(e) (Juang 1994). Apply your algorithm to the discrete-time system model defined by the following data:

$$A = \text{diag}\left(\begin{pmatrix} 0.9859 & 1.500 \\ -1.500 & 0.9859 \end{pmatrix}, \begin{pmatrix} 0.9859 & 0.1501 \\ -0.1501 & 0.9859 \end{pmatrix}, \begin{pmatrix} 0.6736 & 0.7257 \\ -0.725 & 0.6736 \end{pmatrix}, \begin{pmatrix} 0.4033 & 0.9025 \\ -0.9025 & 0.4033 \end{pmatrix}\right).$$

$$B = \begin{pmatrix} -0.0407 & -0.0454 \\ -0.5384 & -0.6001 \\ 0.0746 & -0.0669 \\ 0.9867 & -0.8850 \\ 0.0164 & 0.0373 \\ 0.0376 & 0.0860 \\ -0.0460 & -0.0421 \\ -0.0711 & -0.0650 \end{pmatrix}, \qquad C^{\mathrm{T}} = \begin{pmatrix} 0.8570 & 1.80 \\ 0.0000 & 0.00 \\ 1.5700 & -1.2 \\ 0.0000 & 0.00 \\ 1.4030 & 1.42 \\ 0.0000 & 0.00 \\ 0.9016 & 1.78 \\ 0.0000 & 0.00 \end{pmatrix},$$

$$D = \begin{pmatrix} 0. & 0. \\ 0. & 0. \end{pmatrix}$$

by calculating 200 frequency data points equally spaced in a data frequency ranging from 0 to 16.67 Hz, and assuming that the orders of $Q(z_k)$ and $R(z_k)$ are 10. Sketch the graphs of the true and estimated frequency response functions for the first input and first output and compare the results.

9.11 Consider the following discrete-time model (of a rigid body of mass m with a force f acting along the direction of the motion (Juang 1994):

$$x_{k+1} = Ax_k + Bu_k,$$
$$y_k = Cx_k,$$

where $A = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$, $B = \begin{pmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{pmatrix}$, $u_k = \dfrac{f}{m}$, $C = (1, 0)$.

Δt = sampling time interval.

(a) Construct the first five Markov parameters.

(b) Apply Algorithm 9.3.2 to identify A, B, and C.

(c) Show that the original and the identified models have the identical Markov parameters.

9.12 Using the notation of Section 9.4.1, prove that $\text{span}_{\text{row}}(X_2) = \text{span}_{\text{row}}(H_{k|k+i}) \cap \text{span}_{\text{row}}(H_{k+1|k+2i})$.

9.13 Modify Algorithm 9.4.2 by incorporating weighting matrices W_1 and W_2 such that W_1 is of full rank and W_2 has the property that $\text{rank}(W_{0|i-1}) = \text{rank}(W_{0|i-1}W_2)$.

References

Åström K. and Eykhoff P. "System identification—A survey," *Automatica*, Vol. 7, pp. 123–167, 1971.

Brockett R. "The geometry of the partial realization problem," *Proc. IEEE Conf. Dec. Control*, pp. 1048–1052, 1978.

Datta K.B. "Minimal realization in companion forms," *J. Franklin Institute,* Vol. 309(2), pp. 103–123, 1980.

DeJong L.S. "Numerical aspects of recursive realization algorithms," *SIAM J. Control. Optimiz.*, Vol. 16(4), pp. 646–660, 1978.

DeMoor B., Van Overschee P., and Favoreel W. "Algorithms for subspace state-space systems identification: an overview," *Appl. Comput. Control, Signals, Circuits* (Datta B.N., et al., eds.), vol. 1, pp. 247–311, Birkhauser, Boston, MA, 1999.

Dickinson B., Morf M., and Kailath T. "A minimal realization algorithm for matrix sequences," *IEEE Trans. Autom. Control*, Vol. AC-19(1), pp. 31–38, 1974a.

Dickinson B., Kailath T., and Morf M. "Canonical matrix fraction and state space descriptions for deterministic and stochastic linear systems," *IEEE Trans. Autom. Control*, Vol. AC-19, pp. 656–667, 1974b.

Eykhoff P. *System Identification*, Wiley, London, 1974.

Glover K. "All optimal Hankel-norm approximation of linear multivariable systems and their L_∞-error bounds," *Int. J. Control*, Vol. 39, pp. 1115–1193, 1984.

Gragg W. and Lindquist A. "On the partial realization problem," *Lin. Alg. Appl.*, Vol. 50, pp. 277–319, 1983.

Ho B.L. and Kalman R.E. "Efficient construction of linear state variable models from input/output functions," *Regelungstechnik*, Vol. 14, pp. 545–548, 1966.

Jer-Nan Juang, *Applied System Identification*, Prentice Hall, Englewood Cliffs, NJ, 1994.

Kalman R.E., Falb P.L., and Arbib M.A. *Topics in Mathematical System Theory,* McGraw Hill, New York, 1969.

Kung S.Y. "A new identification method and model reduction algorithm via singular value decomposition," 12th *Asilomar Conf. Circuits, Syst. Comp.*, pp. 705–714, Asilomar, CA, 1978.

Lindquist A. and Picci G. "On subspace methods identification," *Proc. Math. Theory Networks Syst.*, Vol. 2, pp. 315–320, 1993.

Lindquist A. and Picci G. "On subspace methods identification and stochastic model reduction," *Proc. SYSID*, Vol. 2, pp. 397–404, 1994.

Ljung L. *System identification-Theory for the User*, Prentice Hall, Englewood Cliffs, NJ, 1987.

Ljung L. Issues in system identification, *IEEE Control Syst.*, Vol. 11(1), pp. 25–29, 1991a.

Ljung L. *System Identification Toolbox For Use with Matlab*, The Mathworks Inc., MA, USA, 1991b.

McKelvey T. *On State-Space Models in System Identification*, Thesis no. 447, Department of Electrical Engineering, Linköping University, Sweden, 1994a.

McKelvey T. An efficient frequency domain state-space identification algorithm, *Proc.* 33rd *IEEE Conf. Dec. Control*, pp. 3359–3364, 1994b.

McKelvey T. *SSID-A MATLAB Toolbox for Multivariable State-Space Model Identification*, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 1994c.

Moonen M., DeMoor B., Vandenberghe L., and Vandewalle J., "On and off-line identification of linear state space models," *Int. J. Control*, Vol. 49(1), pp. 219–232, 1989.

Norton J.P. *An Introduction to Identification*, Academic Press, London, 1986.

Rissanen J. Recursive identification of linear sequences, *SIAM J. Control*, Vol. 9, pp. 420–430, 1971.

Silverman L. "Realization of linear dynamical systems," *IEEE Trans. Autom. Control*, Vol. AC-16, pp. 554–567, 1971.

Spanos J. T. and Mingori D. L. "Newton algorithm for filtering transfer functions to frequency response measurements," *J. Guidance, Control, Dynam.*, Vol. 16, pp. 34–39, 1993.

Van Overschee P. and DeMoor B. Continuous-time frequency domain subspace system identification, *Signal Processing, Special Issue on Subspace Methods, Part II: System Identification*, Vol. 52, pp. 179–194, 1996a.

Van Overschee P. and DeMoor B. *Subspace Identification for Linear Systems: Theory, Implementation and Applications*, Kluwer Academic Publishers, Boston/London/ Dordrecht, 1996b.

Zeiger H. and McEwen A. "Approximate linear realizations of given dimension via Ho's algorithm," *IEEE Trans. Autom. Control*, Vol. 19, pp. 153, 1974.