156
12

Continuous-Time Multivariable System Identification

by

David L. Cooper, II

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
in
Electrical Engineering

APPROVED:

Stanoje Bingulac, Chairman

William T. Baumann

Hugh F. VanLandingham

September, 1990
Blacksburg, Virginia

Continuous-Time Multivariable System Identification

by

David L. Cooper, II

Committee Chairman: Stanoje Bingulac

Electrical Engineering

(ABSTRACT)

In this thesis, we consider the identification of continuous-time multivariable systems. Direct methods of identification, i.e. identifying a continuous-time model directly from samples of input-output data, are considered briefly. Of primary consideration is the indirect method of identification, which can be considered as a two stage method. First, a discrete system model is identified from samples of input-output data. The next step is to transform this discrete-time model to an equivalent continuous-time representation. The classical Zero-Order hold (ZOH) transformation is presented primarily for comparison with the derived First-Order hold (FOH) technique. Involved in both of these methods is the transformation of the discrete-time state transition matrix to the continuous-time system matrix. A new method for this transformation is presented also. This method along with the presented FOH transformation method have been published in Electronics Letters and another paper on this FOH method has been submitted as an invited paper at the 1991 IFAC Symposium on Identification.

Acknowledgements

I would first like to thank those people who helped make my graduate work a pleasant and memorable experience. More specifically, to the controls lab crew who made work less stressful and more interesting, I remain grateful. To the secretaries, most especially Loretta and Kathy who were always willing to go out of their way to help, I am forever in your debt. And to my friend and colleague James Hutchinson, whose friendship and support were invaluable, I offer my thanks.

I also wish to thank my committee members for their guidance in this project and most of all, my chairman Dr. Bingulac who helped mold this work into a coherent form. My time in graduate school would have been much less challenging and rewarding had I not worked with Dr. Bingulac.

Finally, I would like to thank my parents for their encouragement and support, but most of all, I would like to thank my father, Mr. Lamar Cooper to whom this work is dedicated. Without his help and guidance my graduate career would not have been possible.

Table of Contents

1.0 Introduction

The introduction of digital computers into the field of controls initiated a growing trend toward the use of discrete-time models for the analysis and control of continuous-time systems. Modern control theory has reflected this trend by focusing considerable attention on the development of discrete-time system theory. This move toward discretization has resulted in the derivation of a number of methods for the identification of discrete-time models from samples of input-output data of continuous-time systems.

In many practical situations, however, it may be necessary or desirable to use continuous-time models for several reasons. Several CAD packages, for example, are designed to work exclusively with continuous-time models. Also, methods for analysis and synthesis of continuous-time systems are more familiar to engineers involved in the design of practical control systems, primarily because these systems have been considered much longer than discrete-time systems. Lastly, discrete-time models present difficulties in that they are only valid for the selected sampling interval. These situations point out the need and importance of the deduction of continuous-time models from the existing input-output models.

There are two basic approaches to this problem. The first is the so-called "direct" method. This method obtains the parameters of the continuous-time model directly from samples of input-output data. This method has been considered primarily for SISO systems (Unbenhauen and Rao 1987, Qi-Jie and Sinha 1982) but attempts have been made to extend these methods to the multivariable case (Chen and Hsiao 1975, Sannuti 1977, Wolovich and Guidorzi 1977, Hung, Lui, and Chou 1980).

The second or "indirect" method stipulates the division of the problem into two sub-problems. The first step in this "indirect" method is to identify a discrete-time model of the continuous-time system from samples of input-output data. The second step is to then transform the discrete-time model into an equivalent continuous-time model. The major advantage to this approach is that considerable literature is already available on the first part of the problem (Guidorzi 1975, Bingulac and Krtolica 1985, Bingulac and Farias 1977, Bingulac 1978, Strmčnik and Bremšak 1985, Strmčnik and Tramte 1988, Keviczky and Banyasz 1978, Gorti, Bingulac, and VanLandingham 1990). The second sub-problem has been studied for the single-input single-output case in several papers (Haykin 1972, Sinha 1972, Hsia 1972) and the multivariable (MIMO) system case has been researched in later papers (Strmčnik and Bremšak 1979, Sinha and Lastman 1981, Puthenpura and Sinha 1984, Lastman, Puthenpura, and Sinha 1984, Bingulac and Sinha 1989, Cooper and Bingulac 1990).

In these latter multivariable cases, the continuous-time model is determined from the corresponding discrete-time model based on the assumption that the input signal to the continuous-time system is held constant between sampling intervals. This is a valid assumption if the system is actually sampled using a zero-order hold or if the the sampling interval is sufficiently small so as to validate this assumption. However, for practical cases where the input actually varies between sampling instants, satisfactory results may not be produced from the model obtained under this assumption. For these cases, it may be more reasonable to assume that the input varies linearly between the sampling instants. This is referred to as a "ramp-invariant" or first-order hold (FOH) transformation (Haykin 1972) and has been discussed for single-input single-output (SISO) cases in earlier papers (Sinha 1972, Keviczky 1977) and for multivariable (MIMO) systems in later works (Strmčnik and Bremšak 1979, Bingulac and Sinha 1989, Bingulac and Cooper 1990)

In this thesis, the so-called "indirect" method of continuous-time system identification is primarily considered. The first step of this method, the identification of a discrete-time model based on samples of input-output data, will be presented in Chapter 2. The deterministic identification procedure based on the use of pseudo-observability indices which has recently been proposed by Gorti, Bingulac and VanLandingham (1990) will be reviewed in detail. Chapter 3 will be concerned with the transformation of the identified discrete-time model to an equivalent continuous-time model. The "step-invariant" or zero-order hold (ZOH) tranformation will be considered and compared to a new method of calculating the "ramp-invariant" or first-order hold (FOH) transformation which was recently suggested by Bingulac and Cooper. Chapter 4 contains numerical examples of the primary techniques discussed in Chapters 2 and 3. Conclusions and suggestions for further work are given in Chapter 5. The appendix contains listings of LAS programs which were used to perform the simulations presented in Chapter 4.

2.0 Identification Methods

Identification of system representations is an integral part of the control systems field. Continuous-time system theory developments gave way to discrete-time system theory as the computer became widely used in controls, not only as a device to mechanize numerical calculations, but also as a component in the control system itself. The original continuous-time descriptions of control processes have been widely replaced by discrete approximations and the great number of corresponding developments have tended to obscure parallel developments with continuous-time models, especially in system identification (Unbehauen and Rao 1987).

A large amount of work has been done in the area of identification. Much of the earlier work considered transfer function methods and dealt with SISO systems. With the rise in popularity of the state-space representation, interest has developed in identifying state-space models directly, that is, without considering the transfer functions of the models. This move was also brought about by the need for more simple identification methods for MIMO systems, as

SISO methods which were extended to MIMO cases were rarely straightforward or simple to use.

In this chapter, continuous-time identification methods will be reviewed and a simple method for identifying a discrete-time state-space model from samples of input-output data will be suggested. In section one, direct identification methods for identifying continuous-time models will be considered. Section two will review previous indirect methods or methods for identifying discrete-time models. A recently proposed method involving the use of pseudo-observability indices for structural identification will be covered in section three. At this time, this method seems most promising for identifying discrete-time models of MIMO systems from samples of input-output data.

2.1 Direct Methods

The basic principles behind the various "direct" identification methods for continuous-time models are not very different from discrete-time model methods. These methods can be catagorized into nonparametric and parametric methods. The significance of the term "continuous-time models" is not as strong in the case of nonparametric modelling as in the case of parametric modelling (Unbehauen and Rao 1987). The inherent continuous-time calculus from continuous-time parametric models is eliminated by approximation when they are discretized. Discretization of nonparametric models is primarily for computational convenience and causes less serious consequences. Therefore, a distiction is made between identification methods which estimate points on an unparametrized system model and techniques which estimate a set of parameters in a structured system model.

Methods which estimate impulse and frequency responses are classified as nonparametric methods. Direct impulse input to actual control systems is seldom done due to the potential for damage but seismic and bionic research involves pulse-like signals (Unbehauen and Rao 1987) thereby supporting the need for inpulse response theory. In the area of control engineering, impulse response is normally obtained indirectly from tests using pseudo-random binary signals. Frequency response characterization of dynamic systems has been considered for quite some time in the history of control engineering. These models are relevant to design methods and stability studies. Tthese models are used in analysis of both analog and digital form as well as the Discrete Fourier Transform (DFT) and the Fast Fourier Transform (FFT) which has had a tremendous impact on spectral analysis.

"Direct" identification of parametric models has gained attention as techniques for system design with parametric characterization have been developed. Initially, methods of identifying SISO systems were developed. Later, attempts to extend these methods to MIMO system identification were considered but were in most cases very cumbersome. As of late, most interest lies in multivariable model identification.

Most SISO "direct" parameter estimation methods can be classified in one of two catagories: Output Error (OE) techniques and Equation Error (EE) techniques (Unbehauen and Rao 1987). Figure 1 shows the block diagram of the Output Error configuration. In the OE method, the parameters of an appropriately structured system model are chosen so that they minimize a suitably defined norm of the error between the model output and the actual output of the system to be identified. The scalar OE function in Laplace transform is

$$\mathrm{E}_{OE}(\mathrm{s}) = \mathrm{Y}(\mathrm{s}) - \frac{\mathrm{B}_M(\mathrm{s})}{\mathrm{A}_M(\mathrm{s})}\,\mathrm{U}(\mathrm{s}) \tag{2.1.1}$$

where

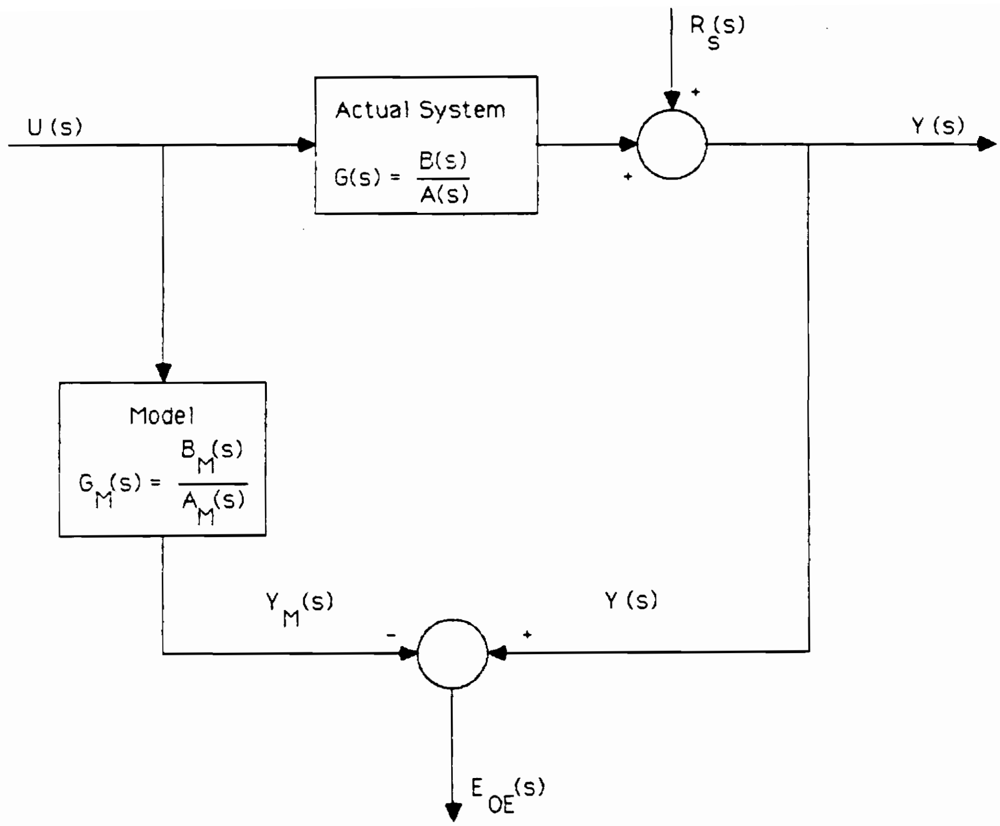

Figure 1. The Output Error (OE) configuration

$$E_{OE}(s) = \text{output error} \tag{2.1.2}$$

$$U(s) = \text{input to the system} \tag{2.1.3}$$

$$Y(s) = \text{output of the system} \tag{2.1.4}$$

and

$$\frac{B_M(s)}{A_M(s)} = \text{model transfer function} \tag{2.1.5}$$

The Equation Error (EE) is generated from the input-output equations of the model and is closely related to the differential approximation concept. Figure 2 shows the block diagram of the Equation Error configuration. The EE is defined as a scalar function given by

$$E_{EE}(s) = A_M(s)\ Y(s) - B_M(s)\ U(s) \tag{2.1.6}$$

where the variables U(s), Y(s), $A_M(s)$, and $B_M(s)$ are given by equations (2.1.3), (2.1.4), and (2.1.5). Equation Error techniques employing orthogonal functions have been well developed. One family of orthogonal systems, consisting of piecewise constant systems of Walsh functions (Chen and Hsiao 1975, Chen et al. 1977) and block-pulse functions (Sannuti 1977) has been employed extensively. Another family uses polynomial systems such as Chebÿshev, Hermite, Legendre, and Laguerre functions. These techniques essentially reduce the calculus of continuous systems to an algebraic approximation in the sense of least squares.

Since these techniques were designed primarily for SISO systems, however, the attempt to extend them to MIMO systems has not been extremely successful. Another problem in extending SISO techniques to MIMO systems is that most of

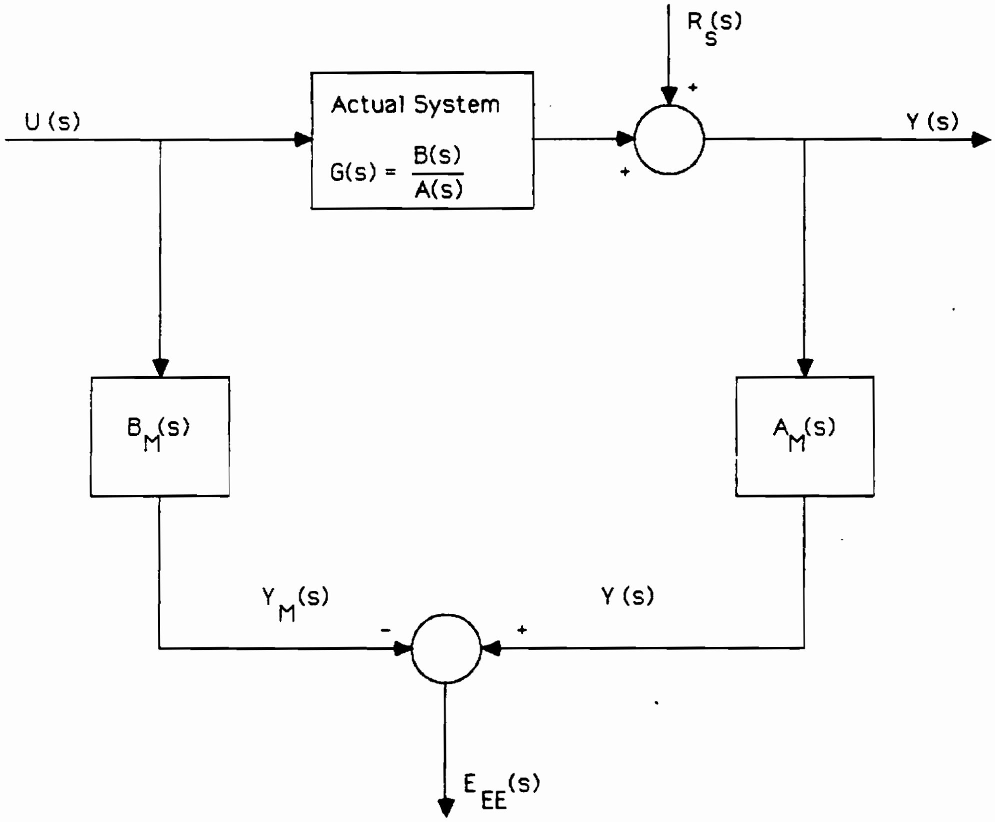

Figure 2. The Equation Error (EE) configuration

these methods consider transfer function equations rather than the more modern state-space realization. Therefore, most of the more recent research in MIMO continuous-time system identification has tended toward state-space model techniques.

A direct procedure for determining continuous-time state-space representations of given time-invariant systems was presented by Wolovich and Guidorzi (1977). Consider the state-space representation

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{2.1.7}$$

$$y(t) = C\ x(t) + E(D)\ u(t) \tag{2.1.8}$$

whose dynamics are expressed in an equivalent differential operator form

$$P(D)\ z(t) = Q(D)\ u(t) \tag{2.1.9}$$

$$y(t) = R(D)\ z(t) + W(D)\ u(t) \tag{2.1.10}$$

where $x \in \mathbf{R}^n$, $u \in \mathbf{R}^m$, $y \in \mathbf{R}^p$, and $z \in \mathbf{R}^q$ are the state, input, output, and partial state vectors respectively. Furthermore, A, B, and C are appropriately dimensioned real matrices while E(D), P(D), Q(D), R(D), and W(D) are appropriately dimensioned polynomial matrices in the differential operator D = d/dt with P(D) being nonsingular. The procedure performs preliminary polynomial matrix operations, if necessary, to reduce the system to an equivalent differential operator form which satisfies the following four specific conditions set by the authors:

1. P(D) is row proper
2. P(D) is column proper
3. the degree of each i^{th} column of R(D) is less than the degree of the corresponding column of P(D)
4. the degree of each j^{th} row of Q(D) is less than the degree of the corresponding row of P(D).

An equivalent state-space representation is then determined by the equations below.

$$A_m = S(D)^{-1} \left(\text{diag} \left[D^{d_i} \right] - P(D) \right) \tag{2.1.10}$$

where S(D) is a specifically structured polynomial matrix of the form

$$S(D) = \begin{bmatrix} s(D) & 0 & \cdots & 0 \\ 0 & s(D) & & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & & s(D) \end{bmatrix} \tag{2.1.11}$$

and where

$$s(D) = [\, 1 \;\; D \;\; D^2 \cdots \; D^{d_i-1}] \tag{2.1.12}$$

To determine matrix B, the equality

$$Q(D) = S^T(D)\, \overline{B} \tag{2.1.13}$$

along with

$$B = M^{-1}\, \overline{B} \tag{2.1.14}$$

is utilized, where M is an (n x n) nonsingular real matrix determined directly from A. Matrix C is solved for directly from

$$R(D) = C\ S(D) \tag{2.1.15}$$

Hung, Lui, and Chou (1980) proposed an algebraic approach using the trapezoidal rule of numerical integration. This technique can be regarded as the combination of the block-pulse function and the inverse trapezoidal rule technique. The development of this technique is as follows. Consider the state-space representation

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{2.1.16}$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the state and input vector respectively, matrices A and B are of compatable dimensions and are to be identified from the measured values of the state vector and input vector at j+1 equally spaced time instants. Integrating equation (2.1.16) yields

$$x_k = A \int_{(k-1)T}^{kT} x\ dt + B \int_{(k-1)T}^{kT} u\ dt + x_{k-1} \tag{2.1.17}$$

where T represents the sampling period. Invoking the trapezoidal rule, (2.1.17) becomes

$$x_k \simeq A\ \frac{x_k + x_{k-1}}{2}\ T + B\ \frac{u_k + u_{k-1}}{2}\ T + x_{k-1} \tag{2.1.18}$$

By rearranging equation (2.1.18), the following equation is developed.

$$\frac{2}{T}(x_k - x_{k-1}) = A\ (x_k + x_{k-1}) + B\ (u_k + u_{k-1}) \tag{2.1.19}$$

By defining the vectors

$$d_k = x_k - x_{k-1} \tag{2.1.20}$$

$$s_k = x_k + x_{k-1} \tag{2.1.21}$$

$$v_k = u_k + u_{k-1} \tag{2.1.22}$$

and the matrices

$$D = [\ d_1 \ \ d_2 \ \dots \ \ d_j] \tag{2.1.23}$$

$$S = [\ s_1 \ \ s_2 \ \dots \ s_j\ ] \tag{2.1.24}$$

$$V = [\ v_1 \ \ v_2 \ \dots \ v_j\ ] \tag{2.1.25}$$

$$C = [\ A \ \ B\ ] \tag{2.1.26}$$

$$Z = \begin{bmatrix} S \\ \cdots \\ V \end{bmatrix} \tag{2.1.27}$$

equation (2.1.19) can be rewritten as

$$\frac{2}{T} D = AS + BV = CZ \tag{2.1.28}$$

If a simple least-square regression is adopted, equation (2.1.28) becomes

$$C = \frac{2}{T} D\ Z^T\ (Z\ Z^T)^{-1} \tag{2.1.29}$$

Qi-Jie and Sinha (1982) point out that this method, although effective, requires the observation of all the state variables as well as the input to obtain the state-space model. These authors propose a modification of this algorithm to use samples of input-output data of a SISO system rather than the states to obtain a state-space model. Since the state-space model is not unique for a given system, the model is assumed to be in the diagonal form. Therefore, let

$$A = \text{diag}\ \{\ \lambda_1, \lambda_2, \dots, \lambda_n\ \} \tag{2.1.30}$$

$$b = [1 \;\; 1 \;\; \dots \;\; 1]^T \tag{2.1.31}$$

$$c = [\; c_1 \;\; c_2 \; \dots \;\; c_n \;] \tag{2.1.32}$$

With this canonical formulation, equation (2.1.18) can be written as

$$\begin{aligned} &\mathrm{diag}[\; (1 - \lambda_1 T/2),\, (1 - \lambda_2 T/2),\, \dots,\, (1 - \lambda_n T\backslash 2) \;]\; x_k \\ &\quad = \mathrm{diag}\; [\; (1 + \lambda_1 T/2),\, \dots,\, (1 + \lambda_n T/2)]\; x_{k-1} + T/2 \;\; b(u_k + u_{k-1}) \end{aligned} \tag{2.1.33}$$

Solving for x_k yields

$$x_k = \mathrm{diag}\; [\; f_1,\, f_2,\, \dots,\, f_n]\; x_{k-1} + \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} (u_k + u_{k-1}) \tag{2.1.34}$$

where

$$f_i = \frac{1 + \lambda_i\; T/2}{1 - \lambda_i\; T/2} \tag{2.1.35}$$

and

$$g_i = \frac{T/2}{1 - \lambda_i\; T/2} \tag{2.1.36}$$

Combining equations (2.1.34) and (2.1.16) leads to an auxiliary discrete-time system with input $(u_k + u_{k-1})$, described by the transfer function

$$H\,(z^{-1}) = \sum_{i=1}^{n} \frac{c_i\; g_i}{1 - f_i\; z^{-1}} \tag{2.1.37}$$

By performing partial fraction expansion on this equation, the values of f_i may be obtained directly. The values of λ_i and g_i may be found by using equations (2.1.35) and (2.1.36). Thus, this previously specified Jordan form state-space representation is found.

2.2 Indirect Methods

The identification of discrete-time models from samples of input-output data system has been given considerable attention, especially since the advent of the digital computer. This relation between the input-output data and the parameters of the systems is usually referred to as an "identification identity" (Bingulac and Farrias 1977). For the single-input single-output case (SISO), the optimal form of the identification identity has already been established (Sinha and Kusta 1983). In the multivariable case (MIMO), work continues on various methods although it has not yet been decided which of the many identification identities is the most convenient.

For multivariable discrete systems, identification methods which treat one output at a time and methods which treat all outputs simultaneously have been considered. Familiarity with single-output systems led researchers to attempt to extend well known SISO structure identification techniques to MIMO systems (Keviczky and Banyasz 1978, Sinha and Kusta 1983, Hsia 1977). These attempts led to unnecessary computationally complex problems.

Many of the earlier attempts at considering all outputs simultaneously in the identification identity have been directed toward the realization of the given unknown system in one of the Luenberger canonical forms (Luenberger 1967). This approach requires that the system structure, (i.e. the observability indices) be determined prior to parameter estimation (Guidorzi 1975, Bingulac and Farrias 1977, Suen and Liu 1978, Strmčnik and Bremšak 1985, Bingulac and Krtolica 1985). The determination of observability indices in this approach is very critical, which is a disadvantage in that if the indices have been determined incorrectly, the parameter estimates are inconsistent.

Since the appearance of pseudo-canonical forms, there has been much interest in the use of pseudo-indices in the identification of multivariable systems. The use of pseudo-canonical forms (i.e. overlapping parameterizations) has removed the prerequisite evaluation of observability indices, thereby allowing both the structure and system parameters to be determined simultaneously (Denham 1974, Glover and Willems 1974, Van Overbeek and Ljung 1982, Wertz, Gevers and Hannan 1982, Correa and Glover 1982, Correa and Glover 1986, Gevers and Wertz 1984). This may be accomplished because the system may be represented by several pseudo-canonical forms. The structure of each of these pseudo-canonical forms is uniquely determined by a corresponding set of structural indices, known as pseudo-observability indices.

Guidorzi (1975) presented a unitary identification procedure for linear discrete multivariable systems based on a preliminary canonical structure identification. This approach requires the direct determination, from the input-output matrix

$$[\ y_1(k)\ y_1(k+1)\ \cdots \vdots \cdots \vdots\ y_m(k)\ \cdots \vdots\ u_1(k)\ \cdots \vdots\ u_r(k)\ \cdots\ ] \tag{2.2.1}$$

of a set of invariant indices, $v_i = \{v_1, ..., v_m\}$, which describe the the input-output structure of the system. This set is determined by selecting the first m independent vectors from the input-output matrix in the following order.

$$y_1(k) \cdots y_m(k),\ u_1(k) \cdots u_m(k),\ y_1(k+1) \cdots y_m(k+1),\ \cdots. \tag{2.2.2}$$

A vector is retained in this selection process if and only if it is independent from previously selected ones. When a dependent vector $y_s(k+v_s)$ is found, all the remaining vectors belonging to the same submatrix will also be dependent so that their test is unnecessary. The selection ends when a dependent vector has been found in every output submatrix. The numbers of vectors selected from these submatrices will be $v_1, ..., v_m$. Now letting

$$L_i(y_i) = [\ y_j(k)\ y_j(k+1)\ \cdots\ y_j(k+i\text{-}1)\ ] \tag{2.2.3}$$

$$L_i(u_i) = [\ u_j(k)\ u_j(k+1)\ \cdots\ u_j(k+i\text{-}1)\ ] \tag{2.2.4}$$

the input-output matrix described by (2.2.1), taking δ_1 vectors in the first submatrix, δ_2 vectors in the second submatrix, etc., can be written as

$$R(\delta_1, ..., \delta_{m+r}) = \{\ L_{\delta_1}(y_1) \ldots L_{\delta_m}(y_m) \vdots L_{\delta_{m+1}}(u_1) \ldots L_{\delta_{m+r}}(u_r)\ \} \tag{2.2.5}$$

A square matrix $S(\delta_1, ..., \delta_m)$ is now defined as

$$S(\delta_1, ..., \delta_m) = R^{T}(\delta_1, ..., \delta_m)\ R(\delta_1, ..., \delta_m) \tag{2.2.6}$$

The vector of parameters, γ_s, is obtained by means of the least squares estimator

$$\gamma_s = (R_s{}^T\ R_s)^{-1}\ R_s{}^T\ y_s(k+v_s) = S_s{}^{-1}\ R_s{}^T y_s(k+v_s) \tag{2.2.7}$$

where

$$R_s = R(v_{s1}, ..., v_{sm}) \tag{2.2.8}$$

$$S_s = S(v_{s1}, ..., v_{sm}) \tag{2.2.9}$$

and where v_{ij} are indices which establish the number of non-zero elements in the state-space form submatrices A_{ij} which are in companion form. Finally a state-space model is easily constructed from the parameter vector.

Bingulac and Farias (1977) and Bingulac (1978) present an algorithm for use in the identification and minimal realization of linear discrete multivariable dynamic systems. The algorithm is based on an identification identity relating input-output data to state-space realization and utilizes the canonical form described earlier by Bingulac (1976). The structural properties of this realization permit the development of an identification identity which treats all outputs simultaneously. The determination of the observability indices and the minimum order of the system is possible with this method. The identification identity is given by

$$G = D\ H \tag{2.2.10}$$

where matrices G and H are defined by

$$G = \begin{bmatrix} y_{i_1}^{\;n_m} \\ \vdots \\ y_{i_p}^{\;n_M} \end{bmatrix} \tag{2.2.11}$$

$$H = \begin{bmatrix} U^{n_M-1} \\ \vdots \\ U^1 \\ U^0 \\ Y^0 \\ Y^1 \\ \vdots \\ Y^{n_M-1} \end{bmatrix} \tag{2.2.12}$$

and matrix D is defined by

$$D = [\, D^{n_M} \,\vdots\, \cdots \,\vdots\, D^2 \,\vdots\, D^1 \,\vdots\, A^* \,] \tag{2.2.13}$$

where n_i represents the observability index with n_m representing the minimum index and n_M representing the maximum index. The submatrix A^* contains the non-zero, non-unity rows of the canonical form system matrix A. Once this identification identity has been solved, submatrices D^i are used to solve for matrix B. Matrix C has an assumed structure of the first p rows of the (n x n) Identity matrix I_n, where n represents the system order and p represents the number of outputs of the system.

Keviczky and Banyasz (1978) consider the extension of several identification methods of SISO linear discrete-time systems to MIMO systems. The methods presented include the Least-Squares (LS) method, the Generalized Least-Squares

(GLS) method, the Maximum Likelihood (ML) method, the Second Extended Matrix (SEXM) method, and the Priori Knowledge Fitting (PKF) method and are given for special forms of the vector difference equation in accordance with the MIMO generalization of the Aström model. The authors chose to identify the parameter matrices of a vector difference equation rather than considering state-space representation since this approach enables the methods considered for SISO systems to be generalized more easily. The vector difference equation used to describe a MIMO linear discrete-time system is given by

$$y(t) = \sum_{i=0}^{n} B_i \, u(t\text{-}i) \; - \; \sum_{i=1}^{n} A_i \, y(t\text{-}i) \; + \sum_{i=1}^{n} C_i \, e(t\text{-}i) \; + e(t) \qquad (2.2.14)$$

where y is a (q x 1) vector of outputs, u is a (m x 1) vector of inputs, and e is a (q x 1) vector of source noise causing corrupting effects at the outputs. It is assumed that the vectors e(t) and e(t+j) are uncorrelated. After the introduction of polynomial matrices $A(z^{-1})$, $B(z^{-1})$, and $C(z^{-1})$, and the introduction of the parameter matrix P, equation (2.2.14) can be written as

$$y(t) = P\, x(t) + C(z^{-1})\, e(t) \qquad (2.2.15)$$

The well known condition of the applicability of the least-squares method is that the equation error $C(z^{-1})$ e(t) should be uncorrelated, but this is fulfilled only if $C(z^{-1}) = I_q$. This condition leads to the familiar equation

$$P = Y\, X^T \, (X\, X^T)^{-1} \qquad (2.2.16)$$

where P is the parameter matrix containing elements of A and B from (2.2.14).

The difference in the general least-squares method is that now $C(z^{-1}) = H^{-1}(z^{-1})$ where $H(z^{-1})\ r(t) = e(t)$. This leads to the similar expression

$$Q = R\ G^T(G\ G^T)^{-1} \tag{2.2.17}$$

where

$$Q = [\ H_1, \ldots, H_k] \tag{2.2.18}$$

$$R = [\ r(1), \ldots, r(N)\ ] \tag{2.2.19}$$

$$G = [\ g(1), \ldots, g(N)\ ] \tag{2.2.20}$$

$$g(t) = [\ -r^T(t\text{-}1), \ldots, -r^T(t\text{-}k)\ ] \tag{2.2.21}$$

and

$$r(t) = A(z^{-1})\ y(t) - B(z^{-1})\ u(t) \tag{2.2.22}$$

The Second Extended Matrix (SEXM) method is quite similar to the least squares technique with the exception that the state vector is now assumed to contain, not only elements of input-output information, but also noise. The noise is estimated as

$$e(t) = y(t) - P\ x(t) \tag{2.2.23}$$

and then the iterative proceedure can be continued as for SISO systems. The Maximum Likelihood (ML) method and the Proiri Knowledge Fitting (PKF) method are quite tedious when applied to MIMO systems and therefore will not be considered here.

Strmčnik and Bremšak (1985) present a recursive algorithm for the transformation of the input-output model into the state-space model which is based on the direct connection of the model parameters in both spaces. To ensure the consistency of the formula, the parameters must be calculated in the correct sequence, which is achieved by using a corresponding index sequence table. Considering the state-space representation

$$x(k+1) = F\ x(k) + G\ u(k) \tag{2.2.24}$$

$$y(k) = H\ x(k) \tag{2.2.25}$$

the state transition matrix F is solved for in the same manner as Guidorzi (1975). Matrix H is assumed on the basis of known indices. The matrix remaining to be solved for is G. The recursive formula derived for the determination of the elements of matrix G is given as

$$g_{jn_i-i+1,h} = \gamma_{ji,h} + \sum_{s=1}^{p} \sum_{t=1}^{n_{j_s}-i} f_{j_s,t+i}\ g_{st,h} \tag{2.2.26}$$

where f_i and γ_i are coefficients of the polynomials of the polynomial matrices P and Q, respectively, which define the input-output relation

$$P(z)\ y(k) = Q(z)\ u(k) \tag{2.2.27}$$

Using equation (2.2.26) for the calculation of the elements of matrix G dramatically reduces the number of calculations needed and therefore reduces computation time.

Bingulac and Krtolica (1985) suggest a new parametrization for identification of discrete-time models of MIMO systems. The parametrization is produced by a specific minimal realization in state-space which can be described by a minimal number of non-zero and non-unity parameters. The input-output model derived from this realization is represented by the same number of parameters and is referred to as the generalized ARMA model. This generalized model is given by

$$\text{DE } y(t) = - \sum_{i=1}^{k} A_i \, y^i(t\text{-}i) + \sum_{i=1}^{k} B_i \, u(t\text{-}i) \tag{2.2.28}$$

where matrix A is in minimal pseudo-canonical form characterized by

$$A = \begin{bmatrix} 0 & & & & & & \\ \vdots & & 1 & & & & \\ a_{11} & \cdots & a_{1r_1} & & & & \\ 0 & & & & & & \\ \vdots & & & & 1 & & \\ a_{j1} & & & \cdots & a_{jr_j} & & \\ 0 & & & & & & \\ \vdots & & & & & & 1 \\ a_{p1} & & & \cdots & & & a_{prp} \end{bmatrix} \tag{2.2.29}$$

matrix B is of the form

$$B = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \tag{2.2.30}$$

The generalized ARMA model presented has fewer parameters as compared to conventional ARMA models and, in deterministic parameter identification based on the least-square approach, leads to well-conditioned matrices.

Strmčnik and Tramte (1988) consider an improvement on an earlier method presented by Strmčnik and Bremšak (1985) which attempts to identify the non-dynamic parts of the system. The improvement lies in the introduction of a simple indexing transformation, allowing the recursive method previously given to be extended to include systems with non-dynamic parts. The transformation of indices used in this identification method is quite subtle but does help identify the non-dynamic parameters of the system more easily. The main advantage pointed out by the authors is that the method calculates the parameters of the state-space model directly from the parameters of the input-output model without the construction of auxiliary matrices. The disadvantage is that it is based on a specific canonical form and is not suitably general.

Bingulac and Krtolica (1988) present an algorithm for simultaneous order identification and parameter estimation of linear, discrete, MIMO systems with unknown observability indices, which may be considered as a multivariable extension of conventional loss function tests used to detect the order of SISO systems. The algorithm identifies the order of the system by checking the linear independence of rows of the matrix

$$\begin{bmatrix} \mathrm{U}(\mu_p;\, \mathrm{N}) \\ \cdots\cdots\cdots \\ \bar{\mathrm{Y}}(\mu_1,..,\, \mu_p;\, \mathrm{N}) \end{bmatrix} \tag{2.2.31}$$

containing the available input-output data, where the rows are arranged according to the assumed values of the pseudo-observability indices $\mu_i = \{\ \mu_1,\ \mu_2,\ \ldots,\ \mu_p\ \}$. Therefore, no previous structure identification is required. The algorithm recursively checks the rank of matrix (2.2.31) until the matrix is of full rank.

Matrix $Y(\mu_1, ..., \mu_p; N)$ is then constructed as

$$Y(\mu_1, ..., \mu_p; N) = \begin{bmatrix} y_1(\mu_1+1) & y_1(\mu_1+2) & \cdots & y_1(\mu_1+N) \\ \vdots & \vdots & & \vdots \\ y_p(\mu_p+1) & y_p(\mu_p+2) & \cdots & y_p(\mu_p+N) \end{bmatrix} \quad (2.2.32)$$

and the following equation solved

$$Y(\mu_1, ..., \mu_p; N) = Q(\mu_1, ..., \mu_p) \begin{bmatrix} U(\mu_p; N) \\ \cdots\cdots\cdots \\ \bar{Y}(\mu_1,.., \mu_p; N) \end{bmatrix} \quad (2.2.33)$$

where the matrix

$$Q(\mu_1, ..., \mu_p) = [\ Q_B \ \vdots \ \ Q_A\ ] \quad (2.2.34)$$

contains the parameters of matrices A and B of the appropriate pseudo-canonical form.

This method was extended by Gorti, Bingulac, and VanLandingham (1990) to derive a computationally simple algorithm for determining a state-space model of a MIMO system from input-output data. The model obtained with this method is in pseudo-observable canonical form. This method does not require structural identification, which is a major advantage over previous identification methods, although it does require the assumption of the system order. The authors show that an identified MIMO system may be represented by a number of possible state-space representations. This number corresponds to the number of admissible sets of pseudo-observability indices. One of these sets leads to the most well conditioned representation. This identification method, which is covered in more detail in the next section, will be utilized in this thesis for the first step in the "indirect" method of continuous-time system identification.

2.3 Pseudo-Observability Indices

Consider a linear time-invariant continuous system

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{2.3.1}$$

$$y(t) = C\ x(t) + D\ u(t) \tag{2.3.2}$$

$$x(0)=x_0 \tag{2.3.3}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $y \in \mathbb{R}^p$, are the state, input, and output vectors respectively, while A, B, C, and D are matrices of compatible dimensions. This system may be represented by an equivalent discrete-time system (Chen 1970, Kailath 1980) which is given by

$$x(k+1) = F\ x(k) + G\ u(k) \tag{2.3.4}$$

$$y(k) = C\ x(k) + H\ u(k) \tag{2.3.5}$$

$$x(0) = x_0 \tag{2.3.6}$$

It is known (Gevers and Wertz 1982, Bingulac and Krtolica 1988, Bingulac and

Krtolica 1987, Correa and Glover 1986) that any n^{th} order MIMO discrete system can be represented by the following pseudo-observable canonical form

$$x(k+1) = F_o\, x(k) + G_o\, u(k) \tag{2.3.7}$$

$$y(k) = C_o\, x(k) + H_o\, u(k) \tag{2.3.8}$$

$$x(0) = x_0 \tag{2.3.9}$$

which is based on a selected set of pseudo-observability indices $\eta_i = \{\eta_1, \eta_2, ..., \eta_p\}$, where p is the number of outputs of the system. It is also know (Bingulac and Krtolica 1987) that this representation is valid only for an admissible set of pseudo-observability indices and that the total number of sets of admissible pseudo-observability indices is less than or equal to

$$I = \frac{(n-1)!}{(p-1)!\,(n-p)!} \tag{2.3.10}$$

According to the admissibility condition specified by Bingulac and Krtolica (Bingulac and Krtolica 1988), the pseudo-observability index that corresponds to a particular component of the output vector should not be greater than its upper bound specified by the individual observability index for this output component.

The pseudo-observable forms of the matrices F_o and C_o are characterized by the following structures.

$$
F_o = \begin{bmatrix}
0 & \cdots & 0 & 1 & & & & \\
f_{11} & \cdots & & f_{1j} & & \cdots & & f_{1n} \\
0 & \cdots & & 0 & 1 & & & \\
f_{21} & \cdots & & f_{2j} & & \cdots & & f_{2n} \\
0 & \cdots & & & 0 & 1 & & \\
 & & & & & & \ddots & \\
0 & \cdots & & & & & 0 & 1 \\
f_{p1} & \cdots & & f_{pj} & & \cdots & & f_{pn}
\end{bmatrix} \tag{2.3.11}
$$

$$
C_o = \begin{bmatrix}
1 & 0 & \cdots & & & & \cdots & 0 \\
0 & 1 & 0 & \cdots & & & \cdots & 0 \\
\vdots & & \ddots & \ddots & & & & \vdots \\
0 & \cdots & 0 & 1 & 0 & \cdots & \cdots & 0
\end{bmatrix} \tag{2.3.12}
$$

It can be seen from the form of (2.3.11) that F_o has only p rows with non-zero and non-unity elements. The location of these rows, $s_i = \{s_1, s_2, ..., s_p\}$, are uniquely determined by the assumed set of pseudo-observability indices η_i. The remaining (n - p) rows of F_o correspond to the last (n - p) rows of the (n x n) Identity matrix I_n. From (2.3.12) it can be noted that the rows in C_o correspond to the first p rows of this Identity matrix. Matrices G_o and H_o have no particular structure, as shown below.

$$
G_o = \begin{bmatrix}
g_{11} & \cdots & g_{1m} \\
\vdots & & \vdots \\
 & & \\
g_{n1} & \cdots & g_{nm}
\end{bmatrix} \tag{2.3.13}
$$

$$H_o = \begin{bmatrix} h_{11} \cdots & h_{1m} \\ \vdots & \vdots \\ h_{p1} \cdots & h_{pm} \end{bmatrix} \tag{2.3.14}$$

The first step in this identification approach is to identify an n^{th} order equivalent discrete-time representation of the n^{th} order continuous-time system (2.3.1) - (2.3.3) from the available input-output sample sequences

$$\{u(k), y(k)\};\ k = 0, 1, ..., N\text{-}1;\ u(k) = u_k;\ y(k) = y_k$$

The discrete representation should satisfy (2.3.7) - (2.3.9), where F_o, G_o, C_o, and H_o are given by (2.3.11) - (2.3.14).

The following equation is obtained from (2.3.7) - (2.3.9).

$$Y(k) = Q_o\, x(k) + R\, U(k) \tag{2.3.15}$$

where the matrices Y(k) and U(k) are (q+1)p and (q+1)p column vectors and Q_o and R are matrices of the dimensions ((q+1)p x n) and ((q+1)p x (q+1)) and where q = n-p+1. The matrix Q_o is the observability matrix calculated from the matrices F_o and C_o given in (2.3.11) and (2.3.14), respectively.

The vectors Y(k) and U(k) have the following structure.

$$Y(k) = \begin{bmatrix} y_k \\ \vdots \\ y_{k+i} \\ \vdots \\ y_{k+q} \end{bmatrix} \qquad U(k) = \begin{bmatrix} u_k \\ \vdots \\ u_{k+i} \\ \vdots \\ u_{k+q} \end{bmatrix} \tag{2.3.16}$$

The matrices Q_o and R have the structure depicted below.

$$Q_o = \begin{bmatrix} C_o \\ C_oF_o \\ \vdots \\ C_oF_o{}^i \\ \vdots \\ C_oF_o{}^q \end{bmatrix} \tag{2.3.17}$$

$$R = \begin{bmatrix} H_o & 0 & \cdots & & & 0 \\ C_oG_o & H_o & 0 & & & \vdots \\ C_oF_oG_o & C_oG_o & \ddots & \ddots & & \\ \vdots & \ddots & & & & \\ & & & & H_o & 0 \\ C_oF_o{}^{q-1}G_o & & \cdots & & C_oG_o & H_o \end{bmatrix} \tag{2.3.18}$$

The elements of the vectors Y(k), U(k), and x(k) are related to the elements in the matrices F_o, G_o, C_o, and H_o by the set of (q+1)p scalar equations described by (2.3.15). The observability matrix Q_o contains n rows equal to the rows in the (n x n) Identity matrix I_n and p rows which contain the non-zero non-unity rows from F_o. As mentioned previously, the location of the rows corresponding to I_n

and the rows corresponding to F_o are uniquely determined by the assumed set of admissible pseudo-observability indices.

Subvectors Y_{1k} and Y_{2k} are now selected from Y_k corresponding to the n rows in Q_o containing rows equal to the rows of I_n and the p rows in Q_o which contain non-zero non-unity rows from F_o, respectively. Submatrices R_1 and R_2 are selected from the corresponding rows of matrix R. Using these submatrices, equation (2.3.15) can be rewritten as

$$Y_{1k} = I\ x_k + R_1\ U_k \tag{2.3.19}$$

$$Y_{2k} = \tilde{F}\ x_k + R_2\ U_k \tag{2.3.20}$$

where

$$\tilde{F} = \begin{bmatrix} f_{11} & f_{1n} \\ \vdots & \vdots \\ f_{p1} & f_{pn} \end{bmatrix} \tag{2.3.21}$$

Solving for x_k in (2.3.19) and substituting this equation into (2.3.20) yields

$$Y_{2k} = \tilde{F}\ Y_{1k} - \tilde{F}\ R_1\ U_k + R_2\ U_k \tag{2.3.22}$$

which may be written in matrix form as

$$Y_{2k} = \begin{bmatrix} \tilde{G} & \vdots & \tilde{F} \end{bmatrix} \begin{bmatrix} U_{1k} \\ \cdots \\ Y_{1k} \end{bmatrix} \tag{2.3.23}$$

where U_{1k} contains the first (η_M+1)m rows from U_k; $\eta_M = \max\ \{\eta_i\}$ for i = 1, ..., p. The matrix $\tilde{G}$ in (2.3.23) is given by $\tilde{G} = R_2 - \tilde{F}\ R_1$.

We now define:

$$\mathrm{Y} = \left[\ \mathrm{Y}_{2k}\ \vdots\ \mathrm{Y}_{2(k+1)}\ \vdots\ \cdots\ \vdots\ \mathrm{Y}_{2(k+q-1)} \right] \tag{2.3.24}$$

$$\mathrm{Z} = \begin{bmatrix} \mathrm{U}_{1k} & \mathrm{U}_{1(k+1)} & \cdots & \mathrm{U}_{1(k+q-1)} \\ \cdots & \cdots & \cdots & \cdots \\ \mathrm{Y}_{1k} & \mathrm{Y}_{1(k+1)} & & \mathrm{Y}_{1(k+q-1)} \end{bmatrix} = \begin{bmatrix} \mathrm{U}_{1i} \\ \cdots \\ \mathrm{Y}_{1i} \end{bmatrix} \tag{2.3.25}$$

where $q \geq n+(\eta_M+1)m$ represents the number of sets of available measurements. The dimensions of Y and Z are (p x q) and $(((\eta_M+1)m+n)$ x q) respectively. It is known that the matrix U_{1i} is of full row rank if the input signal u(k) is sufficiently rich (Bingulac and Farias 1977, Bingulac 1978) and that the assumed set of pseudo-observability indices is, in fact, admissible if matrix Z is of full rank (Bingulac and Krtolica 1988).

Using the Least Squares Method (Sinha and Kuszta 1983), the parameter matrix which depends only on the elements of F_o, G_o, C_o, and H_o may be expressed as

$$\left[\ \tilde{\mathrm{G}}\ \ \vdots\ \ \tilde{\mathrm{F}}\ \right] = \mathrm{Y}\ \mathrm{Z}^T\ (\ \mathrm{Z}\ \mathrm{Z}^T)^{-1} \tag{2.3.26}$$

The matrix F_o is determined directly from (2.3.26) and (2.3.11) while C_o has been previously assumed (2.3.12).

Since F_o and C_o have been determined, G_o and H_o may now be determined. It can be shown that

$$\mathrm{G}_o = \mathrm{Q}_c^{\,e}\ \mathrm{G}^* \tag{2.3.27}$$

where

$$Q_c{}^e = \left[\ G_e \ \vdots \ F_o G_e \ \vdots \ \cdots \ \vdots \ F_o{}^{\eta_M} G_e \right] \tag{2.3.28}$$

which is the $((\eta_M+1)m \times n)$ controllability matrix formed with F_o and the $(n \times p)$ "equivalent" input matrix G_e which contains p rows from the $(p \times p)$ Identity matrix I_p at locations where F_o has non-unity non-zero rows. All other rows in G_e contain only zero elements. The $((\eta_M+1)p \times m)$ matrix G^* is defined by partitioning the $((\eta_M+1)m \times p)$ matrix $\tilde{G}$ into $(p \times m)$ submatrices in the following form.

$$\tilde{G} = \left[\ \tilde{G}_0 \ \vdots \ \tilde{G}_1 \ \vdots \ \cdots \ \vdots \ \tilde{G}_{\eta_M} \ \right] \tag{2.3.29}$$

and

$$G^* = \begin{bmatrix} \tilde{G}_0 \\ \tilde{G}_1 \\ \vdots \\ \tilde{G}_{\eta_M} \end{bmatrix} \tag{2.3.30}$$

To determine H_o, consider the definition of the transfer function matrix of the system described by (2.3.7) - (2.3.9).

$$W(z) = C_o\,(Iz - F_o)^{-1}\,G_o + H_o \tag{2.3.31}$$

The polynomial matrix W(z) is defined by

$$W(z) = P^{-1}(z)\,N(z) \tag{2.3.32}$$

where N(z) and P(z) are the $(p \times m)$ and $(p \times p)$ co-prime polynomial matrices

(Chen 1984), respectively, defined by

$$N(z) = \sum_{i=0}^{n_M} N_i z^i \tag{2.3.33}$$

$$P(z) = \sum_{i=0}^{n_M} P_i z^i \tag{2.3.34}$$

Rewriting (2.3.31) yields

$$H_o = P^{-1}(z)\, N(z) - C_o\, (Iz - F_o)^{-1}\, G_o \tag{2.3.35}$$

Since H_o is independent of the Z-transform variable z, it could be calculated for any arbitrary value of z. If the value of z is chosen to be zero, then P_o and F_o are always non-singular, where P_o is given by the first p columns of $\tilde{F}$. Also, it can be shown (Chen 1984) that $N_o = \tilde{G}_o$. This replacement of the "pseudo-observable" forms of P(z) and N(z), $P_o(z)$ and $N_o(z)$ respectively, and z=0 into (2.3.35) yields

$$H_o = P_o^{-1}\, N_o + C_o\, F_o^{-1}\, G_o \tag{2.3.36}$$

which is the direct calculation for H_o.

An algorithm to calculate the selector vectors s, s_c, h, and r, which specify respectively:

a) the location of non-zero non-unity rows in F_o,

b) the location of the last p rows of I_n in F_o,

c) the location of the rows of a_j in the observability matrix Q_o, and

d) the location of the n rows of I_n in the observability matrix Q_o

has been given in (Gorti, Bingulac, VanLandingham 1990) and is represented below.

1. Define a set $\eta = \{ \eta_1, ..., \eta_p \}$, of admissible pseudo-observability indices, where p is the number of outputs of the system.

2. Set $n = \sum_{i=1}^{p} \eta_i$; n = system order, $\eta_M = \max \{\eta_i\}$, $\eta_P = (\eta_M+1)p$.

3. Set i = 1.

4. Set h = i.

5. For j = 1 through η_P, Set $\overline{V}_x(h) = \eta_i + 1 - j$, h = h + p.

6. Set i = i + 1.

7. If i > p, go to 8; else go to 4.

8. Set $i_1 = 0$, $i_2 = 0$, k = 1.

9. If $\overline{V}_x(k) < 0$, go to 11; else go to 10.

10. Set $i_1 = i_1 + 1$, $\underline{h}(i_1) = k$.

11. If $\overline{V}_x(k) \neq 0$, go to 13; else go to 12.

12. Set $i_2 = i_2 + 1$, $\underline{r}(i_2) = k$.

13. Set k = k + 1.

14. If $k \leq \eta_P$, go to 9; else go to 15.

15. Set q = p + 1, $i_a = 0$, $i_p = 0$, $i_i = 0$.

16. Set $i_p = i_p + 1$.

17. If $\overline{V}_x(q) < 0$, Set $i_p = i_p - 1$ and go to 21; else go to 18.

18. If $\overline{V}_x(q) \neq 0$, Set $i_i = i_i + 1$, else go to 20.

19. Set $\underline{s}_c(i_i) = i_p$.

20. Set $i_a = i_a + 1$, $\underline{s}(i_a) = i_p$, $q = q + 1$.

21. Set $q = q + 1$.

22. if $q < \eta_p$, go to 16; else STOP.

23. Selector vectors $\underline{s}$, $\underline{s}_c$, $\underline{h}$, and $\underline{r}$ are now identified.

Consider, for example, the case where $p = 3$ and $\eta = \{ 1, 4, 2 \}$. The algorithm given above gives the following selector vectors.

$$\underline{s} = \{ 1, 5, 7 \} \qquad \underline{s}_c = \{ 2, 3, 4, 6 \} \tag{2.3.37}$$

$$\underline{h} = \{ 1, 2, 3, 5, 6, 8, 11 \} \qquad \underline{r} = \{ 4, 9, 14 \} \tag{2.3.38}$$

This algorithm, although functional, is designed for implementation and does not give the user any intuition of the form dependency of F_o and Q_o on the assumed set of pseudo-observability indices. The algorithms given here are designed to be simple ways to determine the structures of F_o and Q_o without the need of time consuming algorithm implementation.

To determine the locations of the row types in F_o, perform the following simple steps.

1. First consider the assumed set of pseudo-observability indices $\eta_i = \{\eta_1, \eta_2, ..., \eta_p\}$. Decrement each index in the set η_i and place it in the row vector
$V_1 = \left[(\eta_1\text{-}1) \;\; \vdots \;\; (\eta_2\text{-}1) \;\; \vdots \;\; \cdots\cdots \;\; \vdots \;\; (\eta_p\text{-}1) \right]$.

2. Decrement the elements of V_1 and place them in the row vector V_2. Perform this on each subsequent row vector V_i up to V_{η_M} where $\eta_M = \max \{\eta_i\}$.

3. Now concatonate these row vectors to form $V = [\ V_1 \vdots\ V_2 \vdots \cdots \vdots\ V_{\eta_M}\ ]$ which is a (1 x (η_M*p)) row vector.

4. Maintaining index order, remove indices which are negative from this vector.

5. The remaining vector, $\tilde{V}$, will contain p zeros and (n - p) positive integers. The location of these p zeros in $\tilde{V}$ specify the location in the matrix F_o of the non-zero non-unity rows. The location in $\tilde{V}$ of the (n - p) positive integers specify the location of the last (n - p) rows from the (n x n) Identity matrix I_n in the matrix F_o.

Consider the example given above (equations (2.3.37) and (2.3.38)) for p = 3 and $\eta_i = \{1, 4, 2\}$. Build vector V to obtain

$$V = [\ V_1 \vdots V_2 \vdots V_3 \vdots V_4\ ] = [\ 0\ 3\ 1 \vdots -1\ 2\ 0 \vdots -2\ 1\ -1 \vdots -3\ 0\ -2\ ] \qquad (2.3.39)$$

Removing the negative elements yields

$$\tilde{V} = [\ 0\ 3\ 1\ 2\ 0\ 1\ 0\ ] \qquad (2.3.40)$$

By examination, it can be seen that F_o will contain rows from I_n in rows 2, 3, 4, and 6 and will contain non-zero non-unity elements in rows 1, 5, and 7. This corresponds to

$$\underline{s} = \{\ 1, 5, 7\ \} \text{ and } \underline{s}_c = \{\ 2, 3, 4, 6\ \} \qquad (2.3.41)$$

The determination of the location of the types of rows in Q_o is quite similar to the procedure above. The two differences are:

1) Use the initial set of pseudo-observability indices as V_0 and
2) Do not remove the negative elements of V.

The vector V will contain n positive integers which correspond to the location of the n rows of the (n x n) Identity matrix I_n. Also, p zeros will be contained in V which will correspond to the location of the rows a_j from the matrix F_o.

Consider the previous example with p = 3 and $\eta_i = \{1, 4, 2\}$. Build vector V as before, but include the original set of indices to obtain

$$V = [\ V_0 : V_1 : V_2 : V_3 : V_4\ ] = [\ 1\ 4\ 2 : 0\ 3\ 1 : -1\ 2\ 0 : -2\ 1\ -1 : -3\ 0\ -2\ ] \qquad (2.3.42)$$

It can be seen by inspection that Q_o will contain the n rows from I_n in rows 1, 2, 3, 5, 6, 8, and 11. The rows a_j from F_o are in rows 4, 9, and 14. This corresponds to the selector vectors

$$\underline{h} = \{\ 1, 2, 3, 5, 6, 8, 11\ \} \text{ and } \underline{r} = \{\ 4, 9, 14\ \} \qquad (2.3.43)$$

2.4 Conclusions

In this chapter, numerous methods for identifying continuous-time system have been considered. Direct methods which identify a continuous-model directly from input-output data are available, but for MIMO systems most of these methods are cumbersome to use. This is due to the fact that most of these methods treat only one output at a time and then synthesize these MISO systems. For MIMO systems at the present, indirect methods of identification seem most useful. The response of discrete-time models which are identified using samples of continuous-time input-output data match the response of the continuous-time systems almost perfectly, especially for computationally simple systems.

The discrete identification method presented in section 2.3 is such a method. The use of pseudo-observability indices for structural identification removes the need for seperate structure and parameter identification procedures presented by most identification methods. This identification method shows that the number of possible system representations in identifying MIMO systems corresponds to the number of admissible sets of pseudo-observability indices. Also, one of these sets leads to the most convienient state-space system representation, in that the identified matrices are well conditioned.

3.0 Transforming the Model

As mentioned previously, the second step in the so-called "indirect" method of continuous-time system identification is the transformation of the discrete-time model to an equivalent continuous-time representation. There have been many transformation methods developed for single-input single-output (SISO) systems, most of which were developed as digital filter techniques. These transformation methods include forward and backward rectangular transformations, bilinear and Tustin transformations, with and without prewarping, pole-zero mapping, and Zero-order and First-order hold equivalences. These methods primarily regarded the system transfer function and therefore were not all well suited for extension to the multivariable case.

Of the SISO transformation methods, the step-invariant or Zero-order hold (ZOH), the bilinear, and the ramp-invariant or First-order hold (FOH) transformations have been given the most consideration (Strmčnik and Bremšak 1979, Sinha and Lastman 1982) for multi-input multi-output (MIMO) systems. Previous authors have concluded that of these methods, the bilinear transformation works best for a broader range of system responses (Strmčnik and Bremšak 1979, Sinha and Lastman 1982). Even so, the ZOH method is used in most cases due to the simplicity of implementation and the considerable amount of study which has been directed at this method. The ramp-invariant or first-order hold (FOH) transformation has much potential but has always been computationally challenging for the multivariable case.

In this chapter, some previous research pertaining to transformation techniques is presented in section one. The classical ZOH transformation method is reviewed in section two. The difficulty in this method can be seen to be the calculation of the continuous-time system matrix from the discrete-time state transition matrix, since the calculation of the remaining state-space matrices is straightforward. This problem, which is shown to be the calculation of the natural logarithm of a matrix, is considered in section three. The calculation of this state system matrix is the same for the ZOH and the FOH transformations and therefore it should be considered valid for either transformation method. In section four, a new method of calculating the FOH equivalent system, which was recently proposed by Bingulac and Cooper (1990), is considered. This method has been developed as a more general method than that of Strmčnik and Bremšak (1979) and is less sensitive to zero-valued eigenvalues. It also gives more insight to the ramp-invariant transformation.

3.1 Transformation Techniques

Transformations to the discrete-time domain from the continuous-time domain and vice versa have recieved a large amount of consideration, especially in the case of single-input single-output systems. Considerable work has also been done pertaining to digital filters in connection with the related problem of determining the discrete transfer function corresponding to the transfer function of a continuous filter (Haykin 1967, Sinha 1972). This procedure considers an approximation of the frequency response rather than an approximation of the time-domain response.

Haykin (Haykin 1967) first presented a unified theory of digital filtering which relates the various methods of approximation proposed previously. In review, it was pointed out that the transfer function of a continuous-time filter may be written, in the complex frequency domain, as

$$H(s) = \sum_{k=1}^{N} \frac{A_k}{s + p_k} \tag{3.1.1}$$

An integro-difference equation was developed for deriving the input-output relation of a linear time invariant filter and is given below.

$$y_k(nT) = e^{-p_k T} y_k(nT\text{-}T) + A_k e^{-p_k nT} \int_{nT-T}^{nT} e^{p_k \tau} x(\tau)\, d\tau \tag{3.1.2}$$

The definite integral of (3.1.2) can not be evaluated analytically since the excitation $x(\tau)$ is generally unknown. In order to evaluate equation (3.1.2), some approximation of $x(\tau)$ over the sampling interval $nT\text{-}T < \tau \leq nT$ must be made. For an impulse approximation, $x(\tau)$ is approximated by

$$x(\tau) \simeq T\, x(nT)\, \delta(\tau - nT), \quad nT\text{-}T < \tau \leq nT \tag{3.1.3}$$

which leads to the discrete transfer function

$$H(z^{-1}) = T \sum_{k=1}^{N} \frac{A_k}{1 - e^{-p_k T} z^{-1}} \tag{3.1.4}$$

This, and the following discrete-time transfer functions are the actual analog-to-digital filter transformations which correspond to the approximation of $x(\tau)$. For a step-invariant approximation, $x(\tau)$ is assumed to be

$$x(\tau) \simeq m\, x(nT) + (1\text{-}m)\, x(nT - T), \quad nT\text{-}T < \tau \leq nT \tag{3.1.5}$$

The value of the variable m still must be assumed. A value of zero (0) for m yields the familiar Zero-order hold (ZOH) transformation, where $x(\tau) = x(nT - T)$, which is shown below.

$$H(z^{-1}) = \sum_{k=1}^{N} \frac{A_k}{p_k} (1 - e^{-p_k T}) \frac{z^{-1}}{(1 - e^{-p_k T} z^{-1})} \tag{3.1.6}$$

The ramp-invariant transformation, or First-order hold (FOH) requires the approximation of $x(\tau)$ to be

$$x(\tau) \simeq x(nT - T) + [x(nT) - x(nT - T)] \frac{\tau - (nT - T)}{T} \tag{3.1.7}$$

which leads to the FOH transfer function transformation

$$H(z^{-1}) = \frac{1}{T} \sum_{k=1}^{N} \frac{A_k}{p_k^{\,2}} \frac{(Tp_k - 1 + e^{-p_k T}) + z^{-1}(1 - e^{-p_k T} - Tp_k e^{-p_k T})}{(1 - e^{-p_k T} z^{-1})} \tag{3.1.8}$$

Sinha (Sinha 1972) considers the approximation methods developed previously by Haykin in more detail and compares the precision of each. Sinha's main point in this paper is to show that the bilinear transformation is a good approximation under a wide range of conditions. The bilinear transformation which was used is given below.

$$H(z^{-1}) = \sum_{k=1}^{N} \frac{T\, A_k(1 + z^{-1})}{2 + T\, p_k - (2 - T\, p_k)\, z^{-1}} \tag{3.1.9}$$

The impulse-invariant, step-invariant, and ramp-invariant transformations compared to this bilinear transformation are for SISO systems.

Keviczky (1977) uses transfer function techniques to derive new transformation algorithms for SISO models in state-space form. The transformations considered are the step-invariant and the ramp-invariant.

Consider the SISO continuous-time system representation given by

$$\dot{x}(t) = A\ x(t) + b\ u(t) \tag{3.1.10}$$

$$y(t) = c\ x(t) + d\ u(t) \tag{3.1.11}$$

and the corresponding SISO discrete-time system description

$$x(k+1) = F\ x(k) + g\ u(k) \tag{3.1.12}$$

$$y(k) = c\ x(k) + h\ u(k) \tag{3.1.13}$$

For the step-invariant and ramp-invariant transformations, the transformation of the state transition matrix F to the system matrix A is the same and is given by

$$A = \frac{1}{T} \ln (F) \tag{3.1.14}$$

Keviczky points out that for A to exist, F must have all its eigenvalues inside the unit circle. Transformation equations for the remaining matrices, namely b and d, were derived and are given below for the step-invariant and ramp-invariant cases. Vector c is assumed to be equal in both the discrete-time and continuous-time cases. For step-invariance,

$$b = \frac{1}{T} \ln (F)\ (F - I)^{-1}\ g = A\ (F - I)^{-1}\ g \tag{3.1.15}$$

$$d = h \tag{3.1.16}$$

and for ramp-invariance,

$$b = \frac{1}{T} (\ln (F))^2\ (F - I)^{-2}\ g = T\ A^2\ (F - I)^{-2}\ g \tag{3.1.17}$$

$$d = h - c\ (F - I)^{-1}\ g \tag{3.1.18}$$

The equations were derived for the SISO case and may not be directly extended to the MIMO case.

The approximations which were previously considered by Sinha and Keviczky (Sinha 1972 and Keviczky 1977) for the single input, single output case were extended by Strmčnik and Bremšak to include the multivariable case (Strmčnik and Bremšak 1979). The step-invariant, ramp-invariant, and the bilinear transformations are considered for the multivariable continuous-time system

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{3.1.19}$$

$$y(t) = C\ x(t) + D\ u(t) \tag{3.1.20}$$

and the multivariable discrete-time system

$$x(k+1) = F\ x(k) + G\ u(k) \tag{3.1.21}$$

$$y(k) = C\ x(k) + H\ u(k) \tag{3.1.22}$$

For the step-invariant and ramp-invariant transformations, as in the SISO case, the transformation to the systems matrix A from the state transition matrix F is the same and is given by equation (3.1.14). The step-invariant transformation or Zero-order hold (ZOH) is the simplest type of transformation presented and is given by

$$B = \frac{1}{T} \ln(F)\ (F - I)^{-1}\ G = A\ (F - I)^{-1}\ G \tag{3.1.23}$$

$$D = H \tag{3.1.24}$$

It is obvious that this is the same calculation as in equation (3.1.15) with the exception that this is now a multivariable system. The ramp-invariant or first-order hold (FOH) becomes much more complicated since the feed through matrix now has to be transformed also. This transformation is given below.

$$B = \frac{1}{T} (\ln (F))^2 (F - I)^{-2} G = T A^2 (F - I)^{-2} G \tag{3.1.25}$$

$$D = H+C(F - I)^{-2} [\ln(F)(F - I)^{-1}-I]G = H+C(F - I)^{-2} [AT(F - I)^{-1}-I]G \tag{3.1.26}$$

The bilinear transformation derived by the authors is

$$A = \frac{2}{T}(F - I) (F + I)^{-1} \tag{3.1.27}$$

$$B = \frac{4}{T} (F + I)^{-2} G \tag{3.1.28}$$

$$D = H - C(F + I)^{-1} G \tag{3.1.29}$$

In all three transformation methods, the matrix C was taken to be the same for the discrete-time and continuous-time systems. The authors concluded that if the spectral norm $\| AT \|$ is less than 0.5 then the bilinear transformation yields the closest approximation to the continuous-time system. This condition corresponds to a similar relationship for the scalar case (Sinha 1972) which must be met for the transformation to converge.

Sinha and Lastman (Sinha and Lastman 1981) consider these transformation equations given by Strmčnik and Bremšak, which work when the state transition matrix can be diagonalized but cause problems when the eigenvalues are real and negative and when the state transition matrix cannot be diagonalized. Using an infinite series expansion, the matrices F and G from the model described by equations (3.1.21) and (3.1.22) are expressed as

$$F = e^{AT} = I + AT + \frac{1}{2!}(AT)^2 + \frac{1}{3!}(AT)^3 + \dots \tag{3.1.30}$$

$$G = \int_0^T e^{AT}\, B\, dt = (\, IT + \frac{1}{2!} AT^2 + \frac{1}{3!} A^2T^3 + \dots \tag{3.1.31}$$

The iterative technique proposed to calculate the log of the matrix F has the restriction that the spectral radius, or norm, of AT must be less than 0.5 for the technique to converge, i.e.

$$\|\, AT \,\| < 0.5 \tag{3.1.32}$$

The fixed point iteration proposed by the authors is given below.

$$(AT)^{(k+1)} = (AT)^{(k)} + F^{-1}\,(F - F^{(k)}) = (AT)^{(k)} + I - F^{-1}\, F^{(k)} \tag{3.1.33}$$

where $(AT)^{(k)}$ is the value of AT at the k^{th} iteration and

$$F^{(k)} = e^{(AT)^{(k)}} \tag{3.1.34}$$

This iterative method requires that the spectral radius of AT and F both be less than one for the algorithm to converge. These conditions will be satisfied if A represents a stable system and if the sampling time is selected so that equation (3.1.32) is satisfied.

The fixed point iteration given by equation (3.1.33) works well if the initial guess is sufficiently close to the final solution. The authors recommend using one of the following equations for the initial guess:

$$(\mathrm{AT})^{(0)} = \frac{1}{2}\,(\mathrm{F} - \mathrm{F}^{-1}) \tag{3.1.35}$$

which is obtained by truncating the Taylor series expansions of e^{AT} and e^{-AT} after three terms or

$$(\mathrm{AT})^{(0)} = 2\,(\mathrm{F} + \mathrm{I})^{-1}\,(\mathrm{F} - \mathrm{I}) \tag{3.1.36}$$

which is derived from the Padė approximation of e^{AT}, i.e.

$$\mathrm{F} = e^{AT} \simeq (\mathrm{I} - \tfrac{1}{2}\,\mathrm{AT})^{-1}\,(\mathrm{I} + \tfrac{1}{2}\,\mathrm{AT}) \tag{3.1.37}$$

After AT has been obtained, equation (3.1.31) may be used to calculate B as

$$\mathrm{B} = \mathrm{R}^{-1}\,\mathrm{G} \tag{3.1.38}$$

where

$$\mathrm{R} = [\,\mathrm{I} + \frac{1}{2!}\,\mathrm{AT} + \frac{1}{3!}\,(\mathrm{AT})^2 + \ldots\,]\,\mathrm{T} \tag{3.1.39}$$

Puthenpura and Sinha (Puthenpura and Sinha 1984) proposed applying Chebȳshev polynomials and the Chebȳshev minimax theory to the problem of determining the continuous-time system matrix A from the state transition matrix F. The authors consider the Chebȳshev polynomials on the interval [-1, 1] which are given by

$$T_{r+1}(x) = 2x\, T_r(x) - T_{r-1}(x) \tag{3.1.40}$$

and which correspond to the "shifted" Chebŷshev polynomials on the interval [0, 1]

$$T^*_{r+1}(x) = 2\,(2x - 1)\, T^*_r(x) - T^*_{r-1}(x) \tag{3.1.41}$$

Now the logarithmic series for $x \in [\,0, 1]$ is considered and can be expressed as

$$\ln\,(1 + x) = x - \frac{1}{2}\,x^2 + \frac{1}{3}\,x^3 - \frac{1}{4}\,x^4 + \ldots \tag{3.1.42}$$

which has the property that the error of truncation is smaller than the first neglected term, i.e.

$$\ln\,(1 + x) = x - \frac{1}{2}\,x^2 + \frac{1}{3}\,x^3 - \frac{1}{4}\,x^4 + \ldots + \frac{(-1)^{r-1}}{r} x^r + \epsilon \tag{3.1.43}$$

also

$$\ln\,(1 + x) = p_r(x) + \epsilon \tag{3.1.44}$$

where $p_r(x)$ represents the r^{th}-degree polynomial approximation of $\ln(1 + x)$. Using Chebŷshev polynomials, the lower order approximation may be obtained.

$$p_r(x) = q_s\,[\,p_r(x)\,] + \eta_r \tag{3.1.45}$$

Now using this lower order approximation and equation (3.1.14), the following may be written

$$A = \frac{1}{T} \ln(I + L) = P_r(L) + E_r(L) \tag{3.1.46}$$

where L = F - I. This method uses less number of iterations than the previously proposed iterative technique and has been shown to be especially useful when the discrete-time model of the system is ill-conditioned for inversion.

Lastman, Puthenpura, and Sinha (1984) proposed a new iterative technique which is based on the series given by equation (3.1.46). This summing series can be written as

$$AT = \sum_{k=1}^{\infty} \frac{(-1)^{k+1}\, L^k}{k} \tag{3.1.47}$$

where L = F - I. The error made by approximating AT by $P_r(L)$ is

$$\| E_r(L) \| = \| AT - P_r(L) \| \leq \sum_{k=r+1}^{\infty} \frac{\| L \|^k}{k} \leq \frac{\| L \|^{r+1}}{(r + 1)(1 - \| L \|)} \tag{3.1.48}$$

which shows that this method will converge provided that the spectral radius of L is less than one. This condition is more easily checked than the previous convergence conditions since the matrix F is known. This method seems to be quite efficient computationally, and appears to be more tolerant of multiple and zero-valued eigenvalues than previous methods.

Bingulac and Sinha (1989) consider the ramp-invariant transformation in connection with continuous-time system identification. In this paper, a discrete-time model is identified using samples of input-output data from a continuous-time system. This discrete model is then transformed, via a ramp-invariant transformation, to an equivalent continuous-time system. The problem with the method considered lies in the restrictive assumptions which were made. The restrictions, or assumptions made prior to developing this method were:

1. the order of the model, n, is known
2. the number of outputs, p, is equal to n DIV 2
3. the observability, or pseudo-observability indices, η_i, corresponding to the i^{th} output, is equal to n DIV 2

where the DIV operator represents integer division.

Reconsider the continuous-time and discrete-time models described by equations (3.1.19) - (3.1.22) respectively. With the above assumptions made, the following pseudo-canonical observable form may be selected for the continuous-time model.

$$A_o = \begin{bmatrix} 0 & \vdots & I_{(n-p)} \\ \cdots & \cdots & \cdots \\ F_1 & \vdots & F_2 \end{bmatrix}, \qquad B_o = \begin{bmatrix} B_1 \\ \cdots \\ B_2 \end{bmatrix} \tag{3.1.49}$$

$$C_o = [\ I_p\ \vdots\ 0\ ], \qquad D_o = D \tag{3.1.50}$$

For the ramp-invariant transformation, the authors show that equation (3.1.21) may be rewritten as

$$x(k+1) = F\ x(k) + G_0\ u(k) + G_1\ u(k+1) \tag{3.1.51}$$

In the identification procedure given, first the parameter matrix, P, is found and has the form

$$P = [\ \tilde{B}_1\ \vdots\ \tilde{B}_2\ \vdots\ \tilde{B}_3\ \vdots\ F_1\ \vdots\ F_2\ ] \tag{3.1.52}$$

where

$$\tilde{B}_1 = B_{02} - F_2\, B_{01} - F_1\, D \tag{3.1.53}$$

$$\tilde{B}_2 = B_{12} - F_2\, B_{11} + B_{01} - F_2\, D \tag{3.1.54}$$

$$\tilde{B}_3 = B_{11} + D \tag{3.1.55}$$

The matrices G_0 and G_1 have the form

$$G_0 = \begin{bmatrix} B_{01} \\ \cdots \\ B_{02} \end{bmatrix} \qquad G_1 = \begin{bmatrix} B_{11} \\ \cdots \\ B_{12} \end{bmatrix} \tag{3.1.56}$$

and may be calculated as

$$G_0 = M_0\, B\, T = (ATF - F + I)\ (AT)^{-2}\ BT \tag{3.1.57}$$

$$G_1 = M_1\, B\, T = (F - AT - I)\ (AT)^{-2}\ BT \tag{3.1.58}$$

where

$$M_0 = \sum_{i=0}^{\infty} (i + 1)\, \frac{(AT)^i}{(i + 2)!} \tag{3.1.59}$$

$$M_1 = \sum_{i=0}^{\infty} \frac{(AT)^i}{(i + 2)!} \tag{3.1.59}$$

which lead to the relationship

$$M_0^{-1}\, G_0 = M_1^{-1}\, G_1 \tag{3.1.60}$$

From equations (3.1.53) - (3.1.55), it can be seen that matrix D may be chosen freely. If D is chosen to be zero (0), then one of the other four matrices may be chosen so that equation (3.1.60) is satisfied. Once this is done, matrix B_o is solved for by using one of the equations given below.

$$B = \frac{1}{T} M_0^{-1} G_0 = \frac{1}{T} M_1^{-1} G_1 \tag{3.1.61}$$

Cooper and Bingulac (1990) consider a modification to the iterative technique proposed by Lastman, Puthenpura, and Sinha which causes it to converge more quickly and to be less restrictive on the spectral radius of the matrix L. The improvement involves raising the state transition matrix, F, to the power of 1/2 until the spectral radius L is less than 0.5. This matrix is then passed to the iterative procedure. After the algorithm converges, the resulting matrix is then transformed to the continuous system matrix A by multiplying each element of A by the appropriate power of 2 and by dividing each element by the sampling time. This procedure is computationally simple and has no restriction on distinctness or value of the eigenvalues. This technique will be covered in detail in section 3.3.

A new method of performing ramp-invariant transformation of a discrete-time system to continuous-time domain was proposed by Bingulac and Cooper (1990). The method is constructed in such a way as to be much more general than previous methods. The algorithm presented is valid for any linear system. Section 3.4 considers this procedure, which uses polynomial matrices to reduce the calculation of the ramp-invariant system to the solution of an overdetermined system of linear algebraic equations.

3.2 Zero-Order Hold

Consider a continuous-time linear multivariable system, described by the equations

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{3.2.1}$$

$$y(t) = C\ x(t) + D\ u(t) \tag{3.2.2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$ are the state, input and output vectors, respectively, while A, B, C, and D are matrices of compatible dimensions.

Assuming that the state of the system is known at time $t = t_i$, its value at $t_2 > t_1$ is easily obtained as

$$x(t_2) = e^{A(t_2 - t_1)}\ x(t_1) \quad + \quad \int_{t_1}^{t_2} e^{A(t_2 - \tau)}\ B\ u(\tau)\ d\tau \tag{3.2.3}$$

Now, given the sampling interval as T, the values t_1 and t_2 are selected as

$$t_1 = kT \tag{3.2.4}$$

$$t_2 = (k + 1)T \tag{3.2.5}$$

where k is any positive integer. Equation (3.2.3) can now be written as

$$x_{k+1} = e^{AT}\, x_k \quad + \int_{kT}^{(k+1)T} e^{A(kT+T-\tau)}\, B\, u(\tau)\, d\tau \tag{3.2.6}$$

where $$x_k \triangleq x(kT) \tag{3.2.7}$$

and $$y_k \triangleq y(kT) = C\, x_k + H\, u_k \tag{3.2.8}$$

Equation (3.2.6) can not be integrated in the present form to determine the state of the system and then obtain the output, since the input $u(t)$ is known only at the sampling instants. This problem is simplified considerably if the input to the system is help constant between the sampling instants. This is commonly done in discretizing continuous-time systems for digital control design and implementation. This yields the step-invariant transformation which relates the parameters of the continuous-time system to those of the corresponding discrete-time model. By replacing $u(\tau)$ with u(kT), equation (3.2.6) becomes,

$$x_{k+1} = F\, x_k + G\, u_k \tag{3.2.9}$$

where

$$F = e^{AT} \tag{3.2.10}$$

$$G = \int_0^T e^{A\tau}\, B\, d\tau \tag{3.2.11}$$

and $$u_k = u(kT) \tag{3.2.12}$$

As can be seen by these equations, the problem reduces to the estimation of matrices F, G, C, and H from the samples of input-output data. Matrices A and B may then be computed from the estimates of F and G by using some suitable method. Considering equations (3.2.10) and (3.2.11), equations for the representation of matrices A and B may be found to be

$$A = \frac{1}{T} \ln (F) \tag{3.2.13}$$

and

$$B = A (F - I)^{-1} G \tag{3.2.14}$$

while

$$D = H \tag{3.2.15}$$

and the matrix C in the continuous model is equal to C in the discrete model.

While the calculation of the matrix B using equation (3.2.14) is computationally straight forward, the calculation of the matrix A, or the inverse state transition matrix is not as direct. This problem has been considered by several authors (Sinha and Lastman 1982, Lastman, Puthenpura, and Sinha 1984, Puthenpura and Sinha 1984, Cooper and Bingulac 1990). The next section is concerned with this computation and considers the algorithm which was recently proposed by Cooper and Bingulac.

3.3 The Log of a Square Matrix

Equation (3.2.13), derived from equation (3.2.10), is commonly called the natural log of the matrix F since, for the case where F is a scalar, the scalar variable A can be calculated directly as the logarithm of F divided by T. But this is the trivial case. The calculation of A in the case where F is a square matrix is not quite as direct.

For the case where the matrix F has distinct, non-zero eigenvalues, it can be shown (Sinha and Lastman 1982) that matrix A may be calculated as

$$A = M * \mathrm{diag}\left\{ \frac{1}{T} \ln f_1,\ \frac{1}{T} \ln f_2,\ \dots,\ \frac{1}{T} \ln f_n \right\} * M^{-1} \tag{3.3.1}$$

where the matrix

$$M = \{ v_1, v_2, \dots, v_n \} \tag{3.3.2}$$

and where v_1, v_2, ..., v_n represent the eigenvectors of F corresponding to the eigenvalues f_1, f_2, ..., f_n. The requirement on the eigenvalues of F to be distinct make this method much too restrictive. Special cases are considered by Sinha and Lastman (Sinha and Lastman 1982) but no general formula for calculating the natural log of F is given.

In an earlier work (Sinha and Lastman 1981), a more general iterative method for performing this calculation was given. This method utilized the infinite series definition of the natural logarithm in the calculation. As an improvement to this algorithm, the Chebyshev minimax theory was later used to improve the accuracy of the initial approximation (Puthenpura and Sinha 1984).

One of the latest and most accurate algorithms has been proposed by Lastman, Puthenpura, and Sinha (1984) and is based on the summing series

$$AT = \sum_{k=1}^{\infty} \frac{(-1)^{k+1} L^k}{k} \tag{3.3.3}$$

This algorithm is convergent in the case where $L = F - I_n$ has a spectral radius less than one, where F is the (n x n) state transition matrix and I_n represents the (n x n) Identity matrix. If the spectral radius is between one half and one, i.e. $0.5 \leq \rho(L) \leq 1$, convergence can take numerous iterations causing the algorithm to be painfully slow. Cooper and Bingulac (1990) have proposed an improvement to this algorithm to obtain convergence for spectral radii greater than one and to speed up the iterative process by reducing the spectral radius to or below one half.

To obtain convergence in the case when $\rho(L) > 1$, consider the matrix F as given in equation (3.2.10)

$$F = e^{AT} \tag{3.3.4}$$

which could also be expressed as

$$A = \frac{1}{T} \ln(F) \tag{3.3.5}$$

as in equation (3.2.13). Now let

$$T{=}n{\cdot}t, \tag{3.3.6}$$

leading to

$$t = \frac{T}{n} \tag{3.3.7}$$

where $n = 2^m$, $m = 1, \ldots ,q$. Substituting equation (3.3.6) into equation (3.3.4), we obtain

$$F = e^{An \cdot t} \tag{3.3.8}$$

Now let

$$\check{F} = (F)^{1/2^m} \tag{3.3.9}$$

Consequentially,

$$\check{F} = e^{At} = e^{\check{A}T} \tag{3.3.10}$$

Equation (3.3.5) can now be represented by

$$A = \frac{n}{T} \ln(\check{F}) \tag{3.3.11}$$

leading to

$$\check{A} = \frac{1}{T} \ln(\check{F}) \tag{3.3.12}$$

Rewriting equation (3.3.3) in variables $\check{A}$ and $\check{F}$ leads to

$$\ln(\check{F}) = \check{A}T = \sum_{k=1}^{\infty} \frac{(\check{L})^k}{k}(-1)^{k+1} \tag{3.3.13}$$

where $\check{L} = \check{F} - I$.

This infinite series is convergent, as given in equation (3.3.3), if the spectral radius of $\check{L}$ is less than one. It is also noted that less iterations are needed for smaller spectral radii. This algorithm reduces the spectral radius of the matrix $\check{L}$ sufficiently so as to reduce the number of iterations required by the algorithm, and in the case where the spectral radius is greater than one, reduces the spectral radius so this infinite sum based algorithm will converge. Also, the reduction is limited so as to avoid introducing serious round-off error into the calculations.

The condition to be met is to find a value of m such that the spectral radius of $\check{L} = (\check{F} - I_n)$, where $\check{F}$ is calculated as in equation (3.3.9), is less than 0.5. The algorithm is described as follows.

1. Select a positive integer N and a small positive number ϵ.
2. Set $\check{Q}_1 = \check{L}$ and $\check{S}_1 = \check{L}$ where $\check{L} = (F - I_n)$
3. Set $k = 0$, and $j = 0$
4. a: $j = j + 1$
5. Calculate the spectral radius of the matrix $\check{L}$, $\rho(\check{L})$
6. If $\rho(\check{L}) \leq 0.5$, go to b:
7. If $\rho(\check{L}) > 0.5$, then let $\check{L} = (F^{1/2^j} - I_n)$ and go to a:
8. b: $k = k + 1$
9. Set $\check{Q}_{k+1} = \dfrac{-k\ \check{L}\check{Q}_{k+1}}{(k+1)}$
10. Set $\check{S}_{k+1} = \check{S}_k + \check{Q}_{k+1}$
11. If $k > N$ or $(d(\check{S}_{k+1}, \check{S}_k)) \leq \epsilon$ then go to c:

12. Else, go to b:

13. c: Calculate matrix $A = \check{S} \cdot \frac{1}{T} \cdot 2^{j-1}$

The function d(x,y) can be any suitable measure of the realitive difference or closeness of $\check{S}_{k+1}$ and $\check{S}_k$. One such measure is given by

$$d(\check{S}_{k+1}, \check{S}_k) = \frac{\max_{i,j} | \check{S}_{(k+1)_{i,j}} - \check{S}_{(k)_{i,j}} |}{\max_{i,j} | \check{S}_{(k+1)_{i,j}} |} \tag{3.3.14}$$

The variable N defines the maximum number of iterations carried out in calculating $\check{S}_{k+1}$ and the value of ϵ can be used to control the accuracy of the algorithm.

A possible algorithm to calculate the square root of a general matrix A is given below.

1. Select a small positive number ϵ
2. Set $X_1 = A$, and $i = 0$
3. a: $i = i + 1$
4. Set $X_{i+1} = 0.5(X_i + AX_i^{-1})$
5. If $d(X_{i+1}, X_1) < \epsilon$ then go to b:
6. Else, go to a:
7. b: Set $\sqrt{A} = X_{i+1}$

This algorithm is based on the standard procedure $x_{i+1} = 0.5(x_i + \frac{a}{x_i})$ for calculating the square root, x, of the scalar a where $x = \sqrt{a}$.

This algorithm yields excellent approximations to the matrix A when the spectral radius of $\check{L}$ is less than one. Just as importantly, however, is the fact that in the case where the spectral radius of $\check{L}$ is greater than one, this algorithm will converge to a sufficiently accurate approximation to the continuous-time system matrix.

3.4 First-Order Hold

Given the set of sampled input and output vectors $\{u_j,\ y_j\}$ of a MIMO system, the system can be identified by a discrete-time representation in state space form $\{F,\ G,\ C,\ H\}$ as in chapter 2. The state-space representation of the system is given by

$$x_{k+1} = F\ x_k + G\ u_k \tag{3.4.1}$$

$$y_k = C\ x_k + H\ u_k \tag{3.4.2}$$

where, $x \in \mathbf{R}^n$, $u \in \mathbf{R}^m$, and $y \in \mathbf{R}^p$ are the state, input and output vectors respectively while F, G, C, and H are matrices of compatible dimensions. The sampled input vector is given by : $u_k \triangleq u(kT)$. The system can also be represented by the Z-domain polynomial transfer function equation

$$y(z) = [\ C\ (\ Iz - F)^{-1}\ G + H\ ]\ u(z) \tag{3.4.3}$$

For a step-invariant transformation to a continuous-time model, using Zero-order hold (ZOH) as in section 3.1, there are numerous methods to determine A, B, C, and D (Bingulac and Sinha 1989, Brogan 1985, Sinha and Kuszta 1983). This step-invariant transformation leads to the continuous-time system described by

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{3.4.4}$$

$$y(t) = C\,x(t) + D\,u(t) \tag{3.4.5}$$

where the matrices A, B, C, and D may be calculated from equations (3.2.13) - (3.2.15).

In the step-invariant transformation presented in section 3.2, the input to the system is assumed to be held constant during the sampling interval. This is a valid assumption if the system is actually sampled using a Zero-order hold, but it is not reasonable for a system in which the input to the system is not held constant over the sampling interval. It may be more reasonable to assume the system to have an input which is varying linearly between sampling times, which leads to the following equation for the system input

$$u_{(t)} = u_k \left(\frac{(k+1)T - t}{T}\right) + u_{k+1}\left(\frac{t - kT}{T}\right) \tag{3.4.6}$$

which represents a first-order hold and leads to a ramp-invariant transformation (Bingulac and Sinha 1989, Bingulac and Cooper 1990).

It has been shown (Bingulac and Sinha 1989, Strmčnik and Bremšak 1979) that the state space representation of the discrete system could be written as

$$x_{k+1} = F\,x_k + G_0\,u_k + G_1\,u_{k+1} \tag{3.4.7}$$

$$y_k = C\, x_k + \tilde{D}\, u_k \tag{3.4.8}$$

or in the Z-domain as

$$z \cdot x(z) = F\, x(z) + G_0\, u(z) + G_1\, z\, u(z) \tag{3.4.9}$$

$$y(z) = C\, x(z) + \tilde{D}\, u(z) \tag{3.4.10}$$

where the (n x m) matrices G_0 and G_1 are given by

$$G_0 = M_0 B\, T \quad \text{and} \tag{3.4.11}$$

$$G_1 = M_1 B\, T \tag{3.4.12}$$

Equating (3.4.11) and (3.4.12) we obtain

$$M_1^{-1} G_1 = M_0^{-1} G_o \tag{3.4.13}$$

and solving for G_1 yields

$$G_1 = M_1\, M_0^{-1}\, G_0 \quad \text{or} \tag{3.4.14}$$

$$G_1 = P\, G_0 \tag{3.4.15}$$

where the (n x n) matrix

$$P = M_1\, M_0^{-1} \tag{3.4.16}$$

Matrices M_0 and M_1 are given by

$$M_0 = (\, A\, T\, F - F + I_n\,)\, (\, A\, T)^{-2} \tag{3.4.17}$$

$$M_1 = (\, F - A\, T - I_n\,)\, (\, A\, T)^{-2} \tag{3.4.18}$$

where I_n represents the (n x n) Identity matrix. Similar expressions for matrices G_0 and G_1 in (3.4.7), given by (3.4.7) - (3.4.18) were also derived by Strmčnik and

Bremšak (1979).

Equations (3.4.17) and (3.4.18) cannot be used to calculate M_0 and M_1 if the matrix A is singular, i.e. if the continuous model contains at least one zero eigenvalue, (Bingulac and Sinha 1989). In this case, the equations for M_0 and M_1 are given by

$$M_0 = \sum_{i=0}^{\infty} (i + 1) \frac{(A\ T)^i}{(i + 2)!} \tag{3.4.19}$$

$$M_1 = \sum_{i=0}^{\infty} \frac{(A\ T)^i}{(i + 2)!} \tag{3.4.20}$$

Expressions similar to (3.4.19) and (3.4.20) were derived by VanLandingham and Brogan, among others, in the context of digital filter design (VanLandingham 1985, Brogan 1985).

In order to obtain a continuous time model {A, B, C, D} corresponding to the ramp-invariant transformation, the problem now becomes determining the matrices G_0 and $\tilde{D}$. Note that A is given by either equation (3.2.13) or (3.3.5), B by either equation (3.4.11) or (3.4.12), C is the same for the discrete-time system given in (3.4.2) and the continuous-time system given in (3.4.5), and D is equal to $\tilde{D}$. To solve this problem, first rewrite equation (3.4.9) so that the representations given by equations (3.4.7) & (3.4.8) and (3.4.1) & (3.4.2) describe the same system. Equation (3.4.9) becomes

$$(I_n z - F) x(z) = G_0 u(z) + P G_0 z u(z) \tag{3.4.21}$$

which leads to

$$(I_n z - F) x(z) = (I_n + P z) G_0 u(z) \tag{3.4.22}$$

The Z-domain ploynomial transfer function for this representation can be expressed as

$$y(z) = [C (I_n z - F)^{-1} (I_n + P z) G_0 + \tilde{D}] u(z) \tag{3.4.23}$$

Equation (3.4.23) can also be written as a polynomial matrix

$$y(z) = [C (I_n z - F)_{adj} (I_n + P z) \; \vdots \; \Delta(z) I_p] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} \frac{1}{\Delta(z)} u(z) \tag{3.4.24}$$

where $(X)_{adj}$ represents the adjoint of the matrix X, $\Delta(z) \triangleq$ the characteristic polynomial of the matrix A, and I_p represents the (p x p) Identity matrix. Similarly, equation (3.4.3) can also be written as a polynomial matrix

$$y(z) = [C (I_n z - F)_{adj} G + \Delta(z) H] \frac{1}{\Delta(z)} u(z) \tag{3.4.25}$$

Equations (3.4.24) and (3.4.25) are discrete-time representations of the same system. Therefore, the polynomial matrix equality

$$[\ C\ (\ I_n z - F)_{adj}\ (\ I_n + P\ z\ ) \vdots \Delta(z)\ I_p\ ] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} = [\ C(\ I_n z - F)_{adj}\ G + \Delta(z)\ H] \quad (3.4.26)$$

should hold.

To simplify equation (3.4.26), the (p x m) polynomial matrix

$$W(z) = C\ (\ I_n\ z - F\ )_{adj}\ G \qquad (3.4.27)$$

will be written as

$$W(z) = \sum_{j=0}^{n-1} W_j\ z^j \qquad (3.4.28)$$

where

$$W_j = \begin{bmatrix} w_{11j} & \cdots & w_{1mj} \\ \vdots & \ddots & \vdots \\ w_{p1j} & \cdots & w_{pmj} \end{bmatrix} \qquad (3.4.29)$$

is a (p x m) real matrix containing coefficients w_{ikj} of the polynomials

$$w_{ik}(z) = \sum_{j=0}^{n-1} w_{ikj}\ z^j \qquad (3.4.30)$$

in the polynomial matrix $W(z) = \{w_{ik}(z)\}$. Similarly, the (p x n) polynomial matrix

$$V(z) = C\ (\ I_n\ z - F\ )_{adj}\ I_n \qquad (3.4.31)$$

will be expressed as

$$V(z) = \sum_{j=0}^{n-1} V_j\ z^j \qquad (3.4.32)$$

where

$$V_j = \begin{bmatrix} v_{11j} & \cdots & v_{1mj} \\ \vdots & \ddots & \vdots \\ v_{p1j} & \cdots & v_{pmj} \end{bmatrix} \tag{3.4.33}$$

is a (p x n) real matrix containing coefficients v_{ikj} of the polynomials

$$v_{ik}(z) = \sum_{j=0}^{n-1} v_{ikj}\ z^j \tag{3.4.34}$$

in the polynomial matrix $V(z) = \{v_{ik}(z)\}$.

Substituting equations (3.4.28) and (3.4.32) into equation (3.4.26) yields

$$[\ V(z)\ (\ I_n + P\ z\ ) \vdots \Delta(z)\ I_p] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} = [\ W(z) + \Delta(z)\ H] \tag{3.4.35}$$

The characteristic polynomial, $\Delta(z)$, is assumed to be a monic polynomial of the form

$$\Delta(z) = p_0 + p_1\ z + \ldots + p_{n-1}\ z^{n-1} + 1\ z^n \tag{3.4.36}$$

or

$$\Delta(z) = \sum_{i=0}^{n-1} p_i\ z^i + z^n \tag{3.4.37}$$

Now, substituting equations (3.4.28), (3.4.32), and (3.4.37), equation (3.4.35) becomes

$$\left[\left(\sum_{j=0}^{n-1} V_j z^j \right) (I_n + Pz) \;\vdots\; \sum_{j=0}^{n-1} p_j z^j + z^n I_p \right] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} =$$

$$\left[\sum_{j=0}^{n-1} W_j z^j + \sum_{j=0}^{n-1} p_j H z^j + z^n H \right] \tag{3.4.38}$$

leading to

$$\left[\left(\sum_{j=0}^{n-1} V_j z^j \right) (I_n + Pz) \;\vdots\; \sum_{j=0}^{n-1} p_j z^j + z^n I_p \right] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} =$$

$$\left[\sum_{j=0}^{n-1} \left(W_j + p_j H \right) z^j + z^n H \right] \tag{3.4.39}$$

Equation (3.4.39) can now be expressed as

$$\left[\sum_{j=0}^{n} \left(V_j + V_{j-1} P \right) z^j \;\vdots\; \sum_{j=0}^{n} p_j I_p \; z^j \right] \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} =$$

$$\left[\sum_{j=0}^{n} \left(W_j + p_j H \right) z^j \right] \tag{3.4.40}$$

where $V_{-1} = 0$, $V_n = 0$, $W_n = 0$, and $p_n = 1$.

Combining terms associated with z^j ($j = 0, ..., n$), equation (3.4.36) can formally be written as

$$\left[I_p z^0 \;\vdots\; I_p z^1 \;\vdots\; \cdots \;\vdots\; I_p z^n \right] \left[\tilde{A} \; \tilde{x} \right] = \left[I_p z^0 \;\vdots\; I_p z^1 \;\vdots\; \cdots \;\vdots\; I_p z^n \right] \left[\tilde{B} \right] \tag{3.4.41}$$

where the [(n+1)p x (n+p)], [(n+p) x m], and [(n+1)p x m] matrices $\tilde{A}$, $\tilde{x}$, and $\tilde{B}$ respectively, are given by

$$\tilde{A} = \left[\begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_{n-1} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ V_0 \\ \vdots \\ V_{n-2} \\ V_{n-1} \end{bmatrix} P \begin{array}{c} \vdots\ p_0\, I_p \\ \vdots\ p_1\, I_p \\ \vdots\ \ \vdots \\ \vdots\ p_{n-1} I_p \\ \vdots\ \ I_p \end{array} \right] \tag{3.4.42}$$

$$\tilde{B} = \left[\begin{bmatrix} W_0 \\ W_1 \\ \vdots \\ W_{n-1} \\ 0 \end{bmatrix} + \begin{bmatrix} p_0\, H \\ p_1\, H \\ \vdots \\ p_{n-1}\, H \\ H \end{bmatrix} \right] \quad \text{and} \quad \tilde{x} = \begin{bmatrix} G_0 \\ \cdots \\ \tilde{D} \end{bmatrix} \tag{3.4.43}$$

Equation (3.4.41) will be satisfied for all z if

$$\tilde{A}\tilde{x} = \tilde{B} \tag{3.4.44}$$

Since $(n+1)p \geq n + p$, (the equality holds only if $p = 1$) equation (3.4.44) represents an overdetermined system of linear algebraic equations. The matrices $\tilde{A}$ and $\tilde{B}$ are dependent on the known variables V_i, P, p_i, W_i, and H while the unknown matrix $\tilde{x}$ contains the matrices G_0 and $\tilde{D}$ to be determined. It can be easily verified that $\tilde{B} \subset \mathbf{R}(\tilde{A})$ thus a unique solution exists for this overdetermined system of linear algebraic equations.

Finally, having determined the matrices G_0 and $\tilde{D}$, the ramp-invariant or first-order hold (FOH) equivalent continuous-time representation is given by

$$\dot{x} = A_f\, x(t) + B_f\, u(t) \tag{3.4.45}$$

$$y = C_f\, x(t) + D_f\, u(t) \tag{3.4.46}$$

where

$$A_f = \frac{1}{T} \ln(F) \tag{3.4.47}$$

$$B_f = \frac{1}{T} M_0^{-1} G_0 \tag{3.4.48}$$

$$C_f = C \tag{3.4.49}$$

$$D_f = \tilde{D} \tag{3.4.50}$$

3.5 Conclusions

In this chapter, the Zero-order hold and First-order hold transformations were considered and compared. First, previous work in the area of transformations was presented and discussed. The classical Zero-order hold method was then derived. For MIMO systems, this method is the most popular primarily because of its simple implementation. The most important aspect of the ZOH transformation is the derivation of the continuous-time system matrix A from the discrete state transition matrix F. A method to perform this calculation has been presented in this chapter which appears to currently be the most accurate and most general method, as this procedure has no restrictions on the eigenvalues of matrix A or matrix F. The algorithm, as presented, has no problem transforming even an unstable state transition matrix to a continuous system matrix.

Lastly, a new and exciting method has been presented for calculating a ramp-invariant system representation. This method appears to be the most general method presented to perform this calculation to date. Unlike previous methods, this First-order hold transformation method holds no restrictions on the eigenstructure of the considered continuous-time system. Computationally, this method reduces the deduction of a FOH continuous model to a solution of an overdetermined system of linear algebraic equations having a unique solution.

4.0 Simulation

To illustrate the indirect, continuous-time identification method presented in this thesis, three systems will be considered. The first system, considered in section 4.1, contains a zero eigenvalue and double eigenvalues. This system will be used to show the identification procedure, as presented in section 2.3. This procedure will identify a discrete-time model from which an equivalent continuous-time model can be obtained using a ramp-invariant transformation. Section 4.2 considers a marginally unstable system (i.e. an eigenvalue in the right-half plane and a spectral radius of AT greater than one half) and shows that even under this adverse condition, the procedure for calculating the log of a square matrix, presented in section 3.3, will work satisfactorilly. Last of all, section 4.3 uses the presented procedures to identify a large, random system, primarily to show that the procedures will work for more realistic systems.

The simple steps in the simulation proceedure presented here were as follows:

1. Stimulate the continuous-time system with an input signal and obtain

samples of the system response.

2. Use the samples of the continuous-time system input and system response to identify a discrete-time model .
3. Use the step-invariant and ramp-invariant transformation techniques to transform the discrete-time model to an equivalent continuous-time representation.
4. Compare the accuracy of both procedures.

In the following examples, the ramp-invariant transformation is compared to the classic step-invariant transformation. The responses of both systems are compared to the response of the sampled system to be identified. It can be seen that the ramp-invariant transformation produces a more accurate model than the step-invariant transformation, especially at larger sampling periods.

All calculations and simulations were done either on an IBM PC©, a PC clone or a Sun© workstation in double precision using the computer package LAS© (Bingulac 1988), written by Dr. S. Bingulac, currently at Virginia Tech. Once the simulations were complete, the data was transported to MATLAB© for graphing.

The simulation of a true continuous-time system was accomplished by solving the state-space differential equation at points 0.01 seconds apart. These "samples" were then considered to be the continuous-time input and output data which was sampled at different sample times to identify. The continuous-time models which were calculated were then given the original system input and the response of these identified models then calculated, again every 0.01 seconds.

4.1 Stable Model

The following fourth-order model, with one input and two outputs (n = 4, m = 1, p = 2) was chosen as a continuous-time stable system to be identified

$$A = \begin{bmatrix} -5 & 10 & 0 & 0 \\ 0 & -5 & 10 & 0 \\ 0 & 0 & -1.5 & 6 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \qquad B = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}; \qquad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.1.1}$$

where the matrices A, B, C, and D are defined by

$$\dot{x}(t) = A\ x(t) + B\ u(t) \tag{4.1.2}$$

$$y(t) = C\ x(t) + D\ u(t) \tag{4.1.3}$$

The input to this system is shown in Figure 3. This sinusoidal input was chosen since it is more general than the input signal previously used by Bingulac and Sinha (1989). Sampling times of 0.5, 0.25, and 0.1 seconds were used to

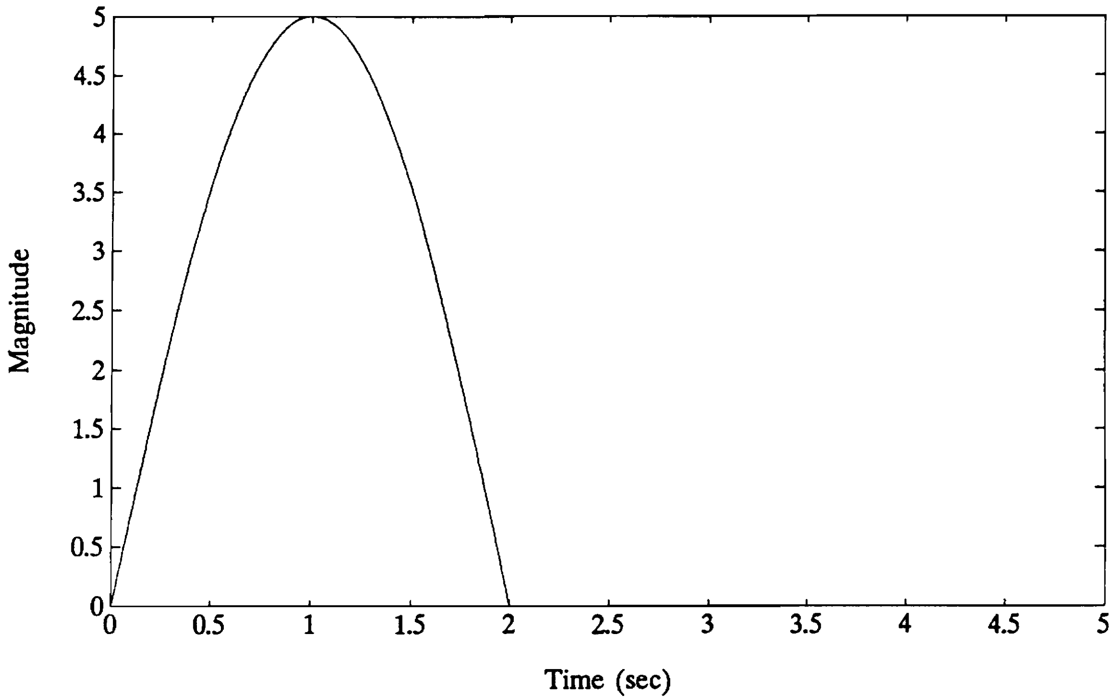

Figure 3. Input Signal u(t)

sample the available input-output data over a 5 second period.

Consider now the observability matrix

$$Q_o = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-p} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ -5 & 10 & 0 & 0 \\ 0 & 0 & -6 & 24 \\ 25 & -100 & 100 & 0 \\ 0 & 0 & 9 & -36 \end{bmatrix} \tag{4.1.4}$$

The possible sets of pseudo-observability indices are $n_1 = \{1, 3\}$, $n_2 = \{2, 2\}$, and $n_3 = \{3, 1\}$. The set of observability indices for Q_o is $n_o = \{2, 2\}$. The set n_1 was determined to be non-admissible. Therefore, for the examples in this section and in section 4.2, the set of pseudo-observability indices given by $n_3 = \{3, 1\}$ were used in the identification proceedure.

The first step is to identify a discrete-time model given by

$$x(k+1) = F_o\, x(k) + G_o\, u(k) \tag{4.1.5}$$

$$y(k) = C_o\, x(k) + H_o\, u(k) \tag{4.1.6}$$

$$x(0) = x_0 \tag{4.1.7}$$

where the matrices F_o, G_o, C_o, and H_o are in pseudo-observable form. In the identification procedure, first the matrices Y and Z were built to use in solving equation (2.3.26) for the parameter matrix

$$[\ \tilde{G} \ \vdots\ \tilde{F}\ ] = Y\, Z^T\, (\, Z\, Z^T)^{-1} \tag{4.1.8}$$

which was calculated as

$$[\ \tilde{G}\ :\ \tilde{F}\ ] = \begin{bmatrix} .011 & -1.213 & -.615 & 0 & .007 & .165 & -.165 & .993 \\ -.018 & 1.166 & 2.155 & .619 & .009 & -.259 & -.223 & 1.473 \end{bmatrix} \tag{4.1.9}$$

leading to

$$F_o = \begin{bmatrix} 0 & 0 & 1 & 0 \\ .007 & .165 & -.165 & .993 \\ 0 & 0 & 0 & 1 \\ .009 & -.259 & -.223 & 1.473 \end{bmatrix} \tag{4.1.10}$$

Using equation (2.3.27), G_o was calculated as

$$G_o = \begin{bmatrix} .586 \\ 1.361 \\ 1.157 \\ 1.771 \end{bmatrix} \tag{4.1.11}$$

Matrix C_o is assumed to be know and given by

$$C_o = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.1.12}$$

Using equation (2.3.36), H_o is calculated as

$$H_o = \begin{bmatrix} .619 \\ 1.730 \end{bmatrix} \tag{4.1.13}$$

The next step in the identification procedure is to transform this discrete-time representation, given by (4.1.10) - (4.1.13). The equivalent continuous-time system matrix A_c is calculated as

$$A_c = \frac{1}{T} \ln(F_o) = \begin{bmatrix} -7.036 & -9.123 & 25.680 & -9.521 \\ .021 & -2.369 & -.477 & 2.825 \\ -.148 & .963 & -3.412 & 2.597 \\ .030 & -.514 & -.886 & 1.370 \end{bmatrix} \qquad (4.1.14)$$

which has eigenvalues at $\lambda_i = \{ -4.973 \pm 0.2i, -1.5, 0.0 \}$.

Since A_c is singular, the matices M_0 and M_1, used in calculating G_0 and G_1 as in equations (3.4.11) and (3.4.12), must be calculated by the power series defined in equations (3.4.19) and (3.4.20). These matrices are given below.

$$M_0 = \begin{bmatrix} .055 & -.112 & .739 & -.181 \\ .003 & .197 & -.060 & .361 \\ -.002 & .028 & .114 & .360 \\ .003 & -.089 & -.087 & .673 \end{bmatrix} \qquad (4.1.15)$$

$$M_1 = \begin{bmatrix} .196 & -.174 & .712 & -.235 \\ .001 & .334 & -.033 & .197 \\ -.003 & .032 & .277 & .194 \\ .002 & -.045 & -.052 & .595 \end{bmatrix} \qquad (4.1.16)$$

To calculate the FOH system, equation (3.4.40) is implemented, solving for G_o and D. Using equations (3.4.48) to (3.4.50), matrices B_f, C_f and D_f of the First-order hold system are calculated and given below.

$$B_f = \begin{bmatrix} 1.104 \\ 4.201 \\ 6.172 \\ 11.285 \end{bmatrix}; \; C_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \; D_f = \begin{bmatrix} .003 \\ .016 \end{bmatrix} \qquad (4.1.17)$$

The response of this ramp-invariant system to the input given by Figure 3 is shown in Figure 4.

For comparison, a step-invariant (ZOH) system is now calculated using equations (3.2.13) to (3.2.15). For this system, matrix A_c in the same as (4.1.14). The remaining matrices are given below.

$$B_z = \begin{bmatrix} 3.336 \\ 7.959 \\ 8.915 \\ 12.888 \end{bmatrix}; \; C_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \; D_z = \begin{bmatrix} .619 \\ 1.730 \end{bmatrix} \tag{4.1.18}$$

Figure 5 shows the response of this ZOH system plotted with the sampled system response.

Now using a sampling time of 0.25 seconds, the output of the system given by (4.1.1) was sampled and a discrete-time representation was obtained. From this model, the matrix A_c was obtained and is given below.

$$A_c = \frac{1}{T} \ln (F_o) = \begin{bmatrix} -9.312 & -3.044 & 16.131 & -3.755 \\ .669 & -4.235 & -4.714 & 8.280 \\ -.916 & .718 & -2.607 & 2.805 \\ .463 & -.720 & -4.374 & 4.631 \end{bmatrix} \tag{4.1.19}$$

which has eigenvalues at $\lambda_i = \{ -5.193, -4.830, -1.5, 0.0 \}$. For the ramp-invariant system, the remaining matrices were calculated as

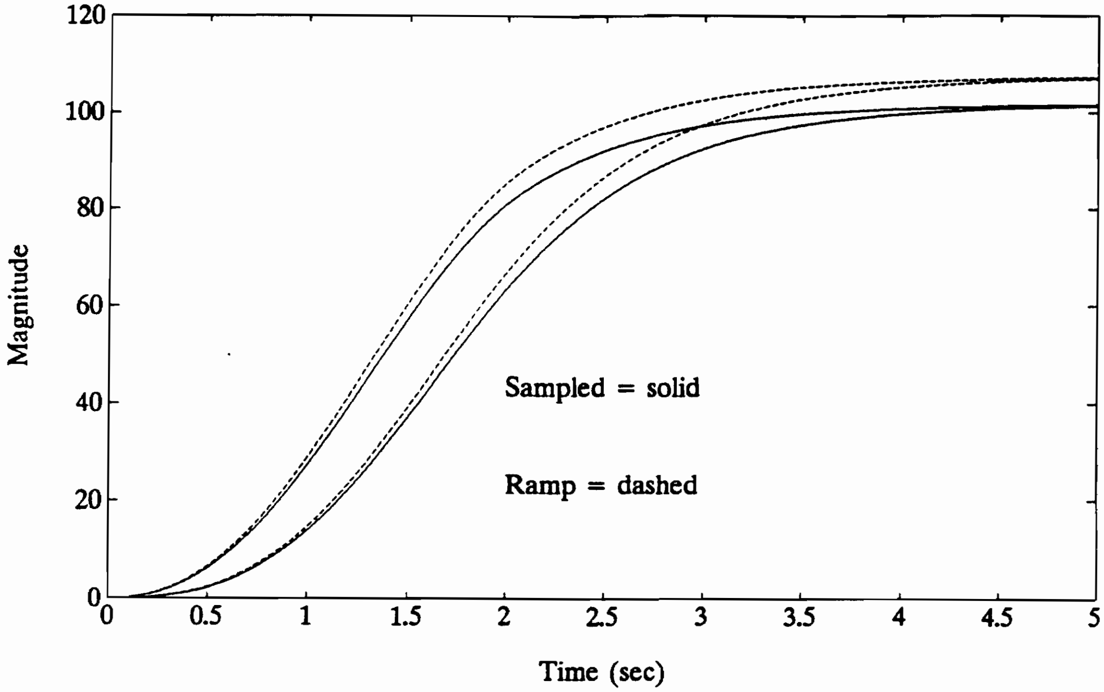

Figure 4. Outputs y(t) of Sampled and Ramp-Invariant Continuous-Time Models; $\Delta t = 0.5$ sec

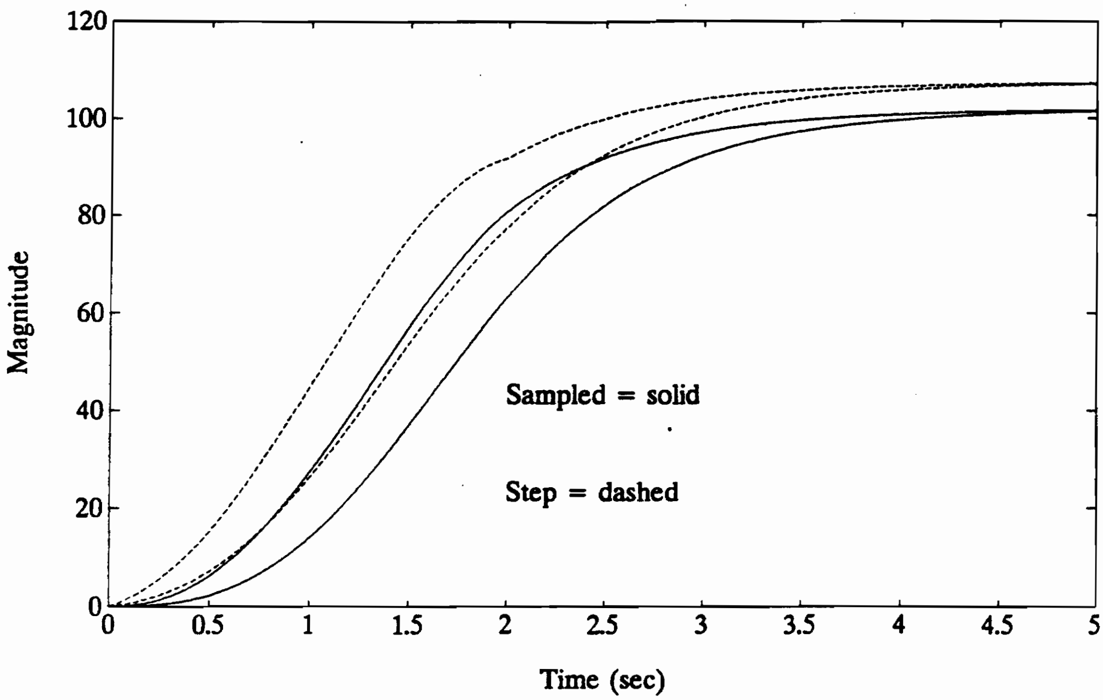

Figure 5. Outputs y(t) of Sampled and Step-Invariant Continuous-Time Models; $\Delta t = 0.5$ sec

$$\mathrm{B}_f = \begin{bmatrix} 1.011 \\ 4.049 \\ 3.053 \\ 5.932 \end{bmatrix}; \mathrm{C}_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \mathrm{D}_f = \begin{bmatrix} 0 \\ .001 \end{bmatrix} \tag{4.1.20}$$

The step-invariant system is given below.

$$\mathrm{B}_z = \begin{bmatrix} 1.913 \\ 6.070 \\ 4.472 \\ 7.333 \end{bmatrix}; \mathrm{C}_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \mathrm{D}_z = \begin{bmatrix} .197 \\ .681 \end{bmatrix} \tag{4.1.21}$$

Figures 6 and 7 show the responses of these ramp-invariant and step-invariant systems, respectively, plotted with the response of the sampled system.

Finally, a sample time of 0.1 seconds was used. The matrix A_c is given below.

$$\mathrm{A}_c = \frac{1}{\mathrm{T}} \ln\left(\mathrm{F}_o\right) = \begin{bmatrix} -9.312 & -3.044 & 16.131 & -3.755 \\ .669 & -4.235 & -4.714 & 8.280 \\ -.916 & .718 & -2.607 & 2.805 \\ .463 & -.720 & -4.374 & 4.631 \end{bmatrix} \tag{4.1.22}$$

which has eigenvalues at $\lambda_i = \{\text{ -5.216, -4.805, -1.5, 0.0 }\}$. The ramp-invariant system was calculated as

$$\mathrm{B}_f = \begin{bmatrix} 1.001 \\ 4.006 \\ 1.634 \\ 2.510 \end{bmatrix}; \mathrm{C}_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; \mathrm{D}_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.1.23}$$

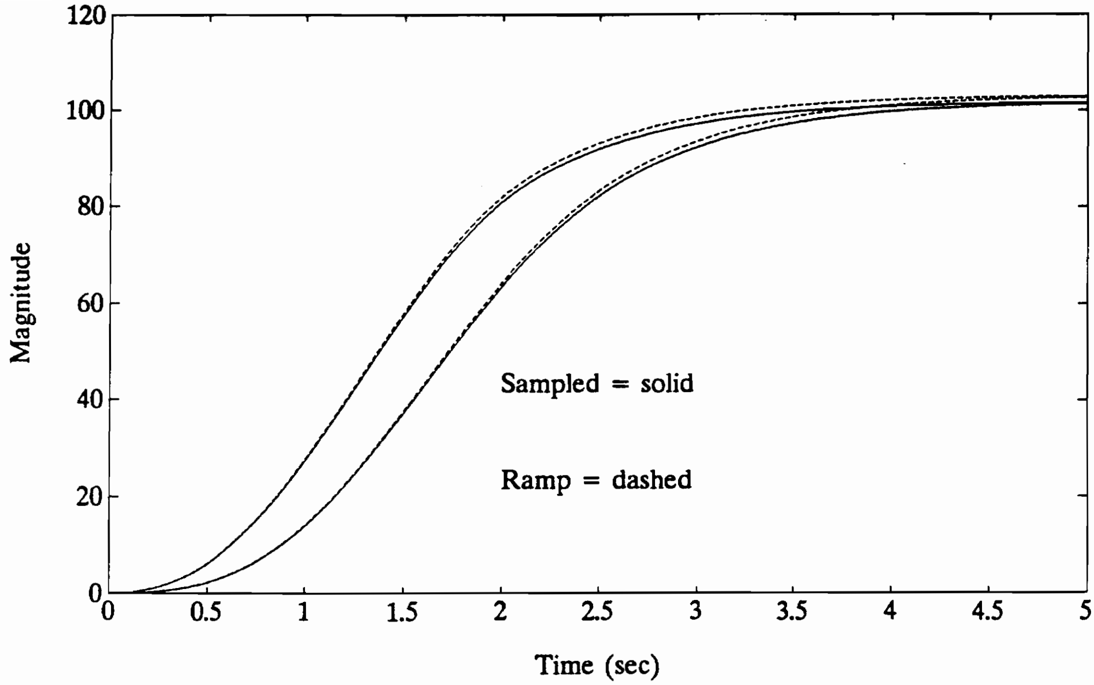

Figure 6. Outputs y(t) of Sampled and Ramp-Invariant Continuous-Time Models; $\Delta t = 0.25$ sec

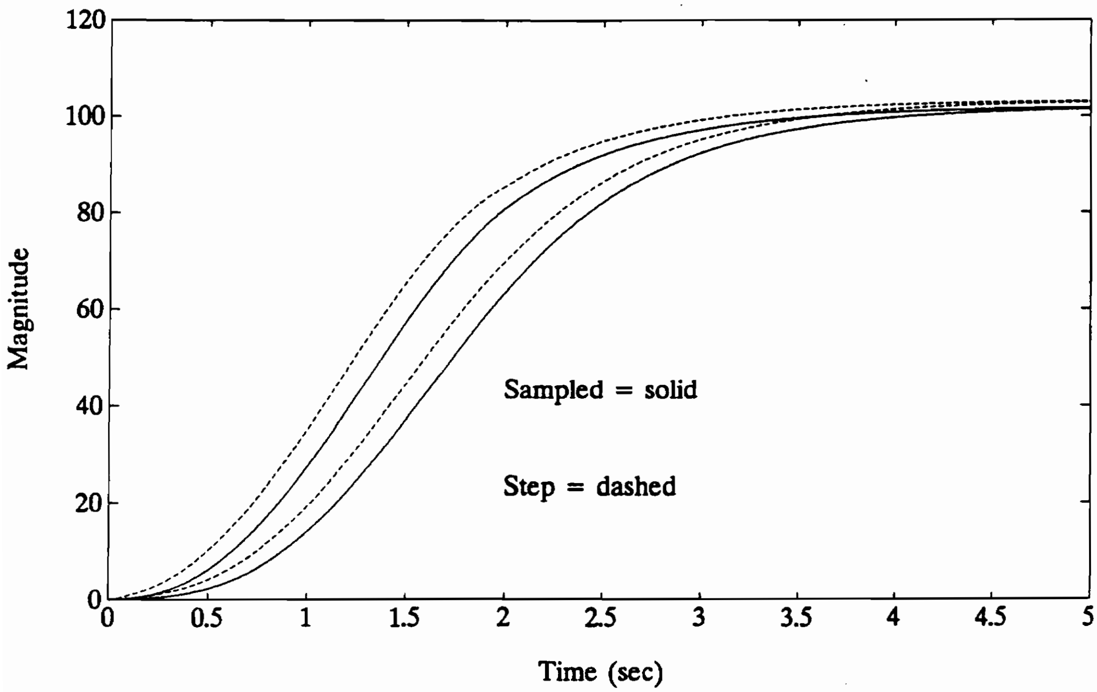

Figure 7. Outputs y(t) of Sampled and Step-Invariant Continuous-Time Models; $\Delta t = 0.25$ sec

and the step-invariant system for this sampling time was found to be

$$B_z = \begin{bmatrix} 1.296 \\ 4.865 \\ 2.054 \\ 3.024 \end{bmatrix}; C_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; D_z = \begin{bmatrix} .060 \\ .230 \end{bmatrix} \qquad (4.1.24)$$

Figures 8 and 9 show the responses of these ramp-invariant and step-invariant systems, respectively, plotted with the sampled system response.

To lend more insight into the accuracy of the ramp-invariant transformation as compared with the classic step-invariant transformation, Figure 10 shows the error of the ramp-invariant system for each sample time, which was calculated as the difference between the response of the sampled system and the response of the step-invariant transformed system response. Since the errors of both output vectors of the multi-output system given in (4.1.1) are approximately equal, only the error of the first output, $y_1(t)$, is shown. Figure 11 shows the error of the step-invariant system for each sample time. It is obvious from these graphs, Figures 10 and 11, that the ramp-invariant system more accurately approximates the original system for every sampling time for this type of input signal.

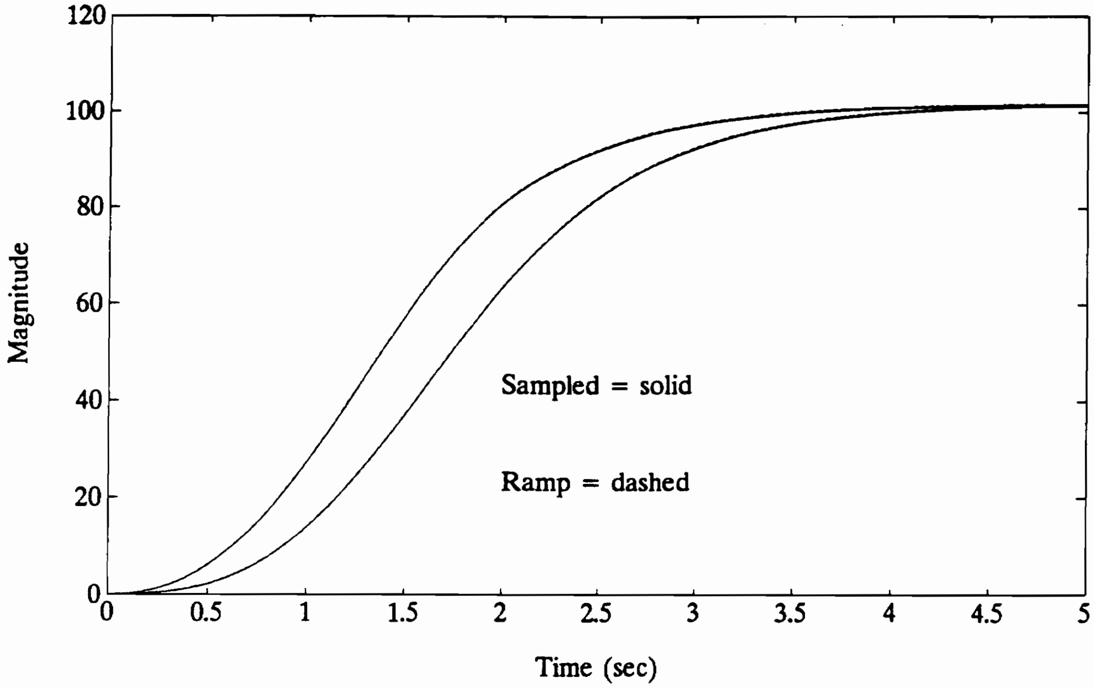

Figure 8. Outputs y(t) of Sampled and Ramp-Invariant Continuous-Time Models; $\Delta t = 0.1$ sec

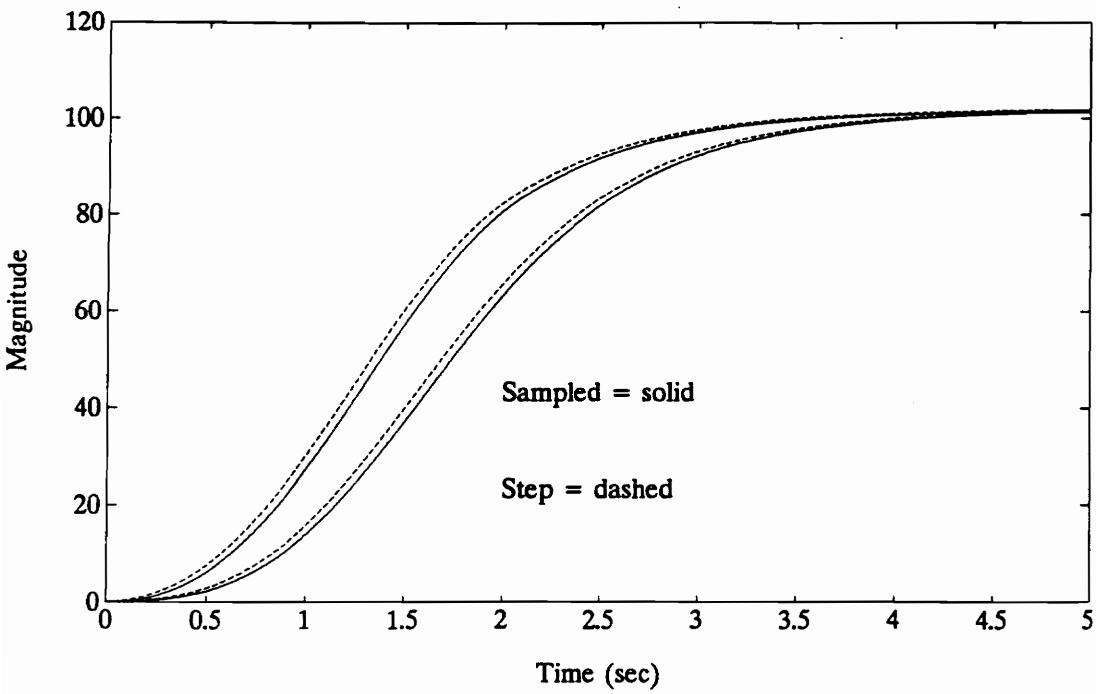

Figure 9. Outputs y(t) of Sampled and Step-Invariant Continuous-Time Models; $\Delta t = 0.1$ sec

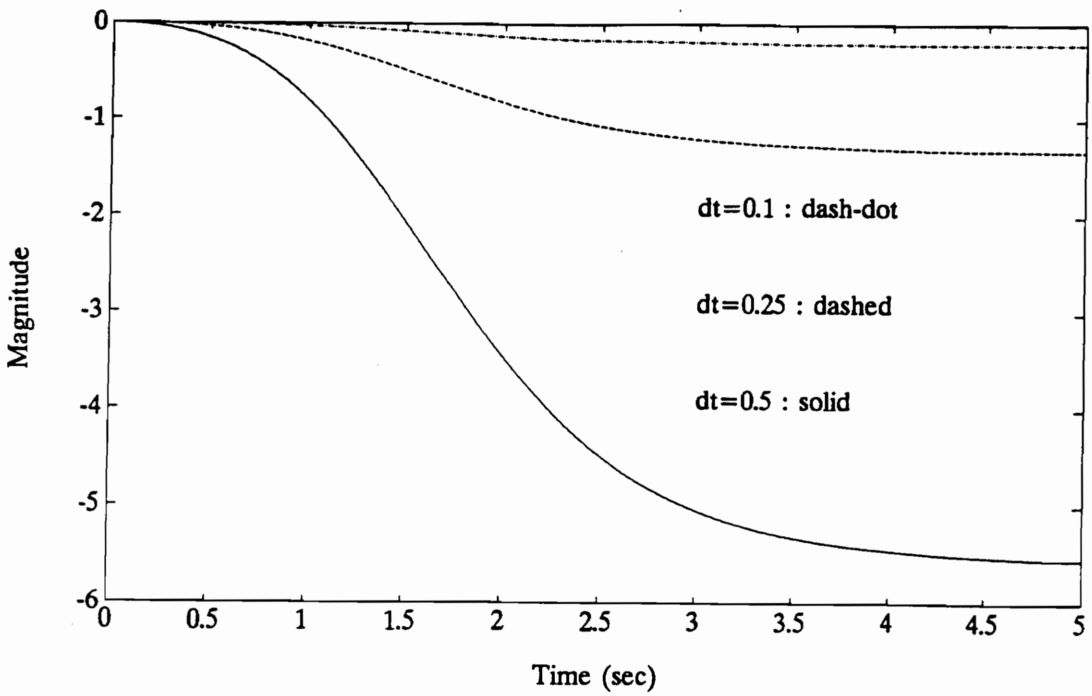

Figure 10. Differences between Sampled and Ramp-Invariant System Responses; $\Delta t = 0.5, 0.25, 0.1$ sec

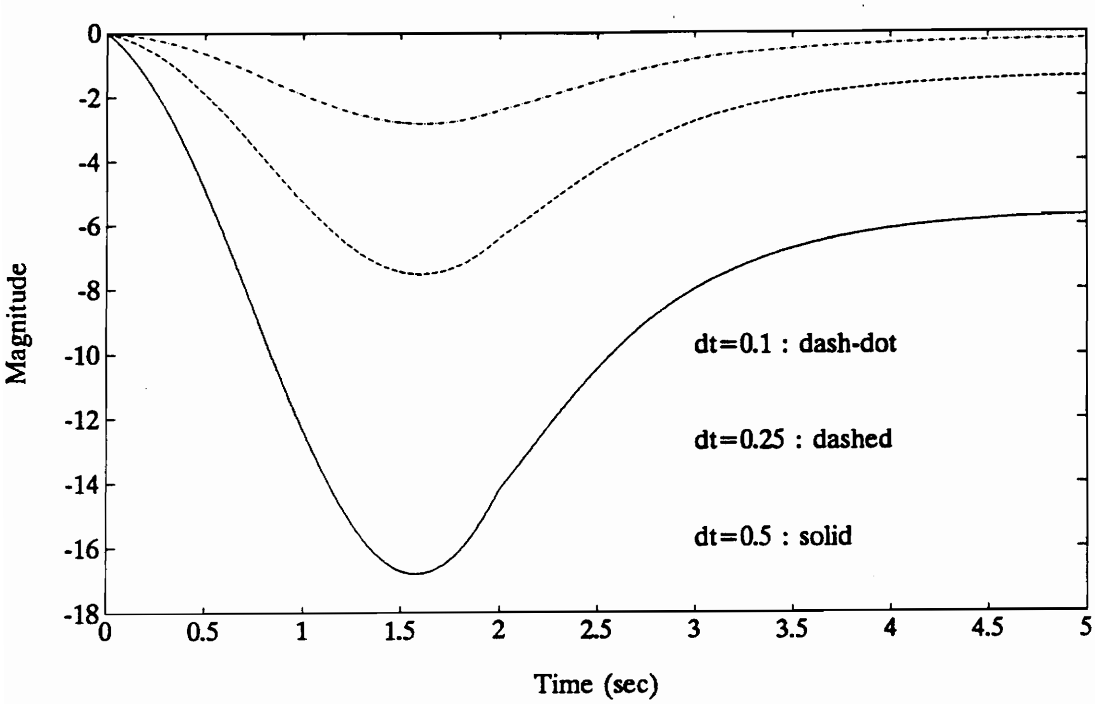

Figure 11. Differences between Sampled and Step-Invariant System Responses; $\Delta t = 0.5, 0.25. 0.1$ sec

4.2 Unstable Model

The purpose of this section is to exercise the method presented in section 3.3, i.e. perform a transformation of the identified state transition matrix to a continuous-time system matrix when the spectral radius of AT, the original sampled continuous-time system matrix multiplied by the sampling time, is greater than one half. As pointed out by Lastman, Puthenpura, and Sinha, an easier constraint to calculate is that the spectral radius of F - I is less than one, since the state transition matrix F is known. Therefore, to actually set up a system which meets this criteria, consider the scalar case, given by

$$\mathrm{f} = \mathrm{e}^{\mathrm{a}T} \tag{4.2.1}$$

For our purposes, it is most desirable to have an eigenvalue which is relatively unstable, but not so unstable so as to present an excessively large response magnitude, which could cause serious rounding error. Therefore, if the unstable eigenvalue is chosen to be

$$\lambda_{us} = 1.5 \tag{4.2.2}$$

and the criteria to meet is

$$f\text{-}1 = e^{aT} - 1 \geq 1 \tag{4.2.3}$$

then a sample time is found to be $\simeq$ 0.46, which will be rounded up to 0.5.

The following fourth-order model was chosen as an unstable continuous-time system to be identified,

$$A = \begin{bmatrix} -5 & 10 & 0 & 0 \\ 0 & -5 & 10 & 0 \\ 0 & 0 & 1.5 & 6 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \qquad B = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}; \qquad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.2.4}$$

where the matrices A, B, C, and D are defined by equations (4.1.2) and (4.1.3).

Using a sampling time of 0.5 seconds and an input signal given by Figure 3, the output of the system given by (4.2.4) was sampled and a discrete-time representation, as given by equations (4.1.5) - (4.1.7), was obtained. The corresponding matrices are

$$F_o = \begin{bmatrix} 0 & 0 & 1 & 0 \\ .004 & .411 & -.109 & .694 \\ 0 & 0 & 0 & 1 \\ .016 & -1.447 & -.432 & 2.863 \end{bmatrix} \tag{4.2.5}$$

$$G_o = \begin{bmatrix} 6.086 \\ 14.991 \\ 21.003 \\ 53.703 \end{bmatrix} \qquad H_o = \begin{bmatrix} .747 \\ 2.699 \end{bmatrix} \tag{4.2.6}$$

where, again

$$C_o = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.2.7}$$

and L = F_o - I_n has a spectral radius of 1.117. From this model, the matrix A_c was obtained and is given below.

$$A_c = \frac{1}{T} \ln (F_o) = \begin{bmatrix} -7.233 & -14.265 & 26.856 & -5.358 \\ .005 & -.792 & -.144 & .931 \\ -.141 & 1.885 & -3.361 & 1.617 \\ .033 & -1.564 & -1.045 & 2.576 \end{bmatrix} \tag{4.2.8}$$

which has eigenvalues at λ_i = { -5.695, -4.616, 1.5, 0.0 }. For the ramp-invariant system, the remaining matrices were calculated as

$$B_f = \begin{bmatrix} .935 \\ 4.249 \\ 10.612 \\ 39.276 \end{bmatrix}; C_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; D_f = \begin{bmatrix} .012 \\ .027 \end{bmatrix} \tag{4.2.9}$$

The step-invariant system is given below.

$$B_z = \begin{bmatrix} 4.524 \\ 14.586 \\ 22.977 \\ 66.611 \end{bmatrix}; C_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}; D_z = \begin{bmatrix} .747 \\ 2.699 \end{bmatrix} \tag{4.2.10}$$

Figures 12 and 13 show the responses of these step-invariant and ramp-invariant systems, respectively, plotted with the response of the sampled system.

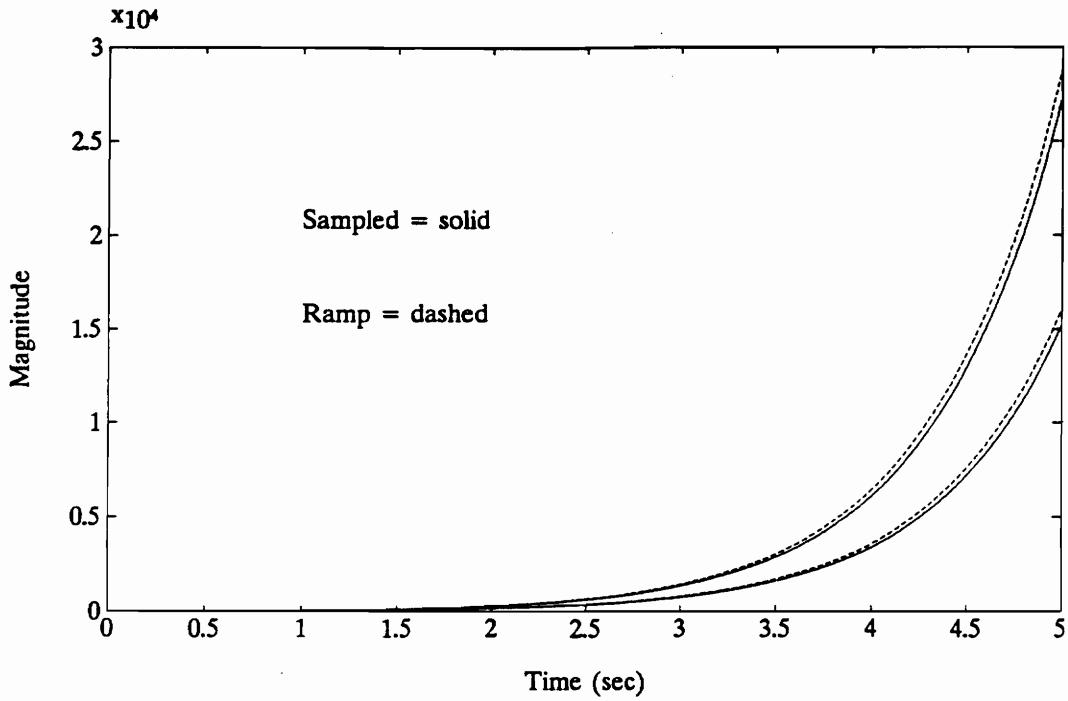

Figure 12. Outputs y(t) of Sampled and Ramp-Invariant Continuous-Time Models; $\Delta t = 0.5$ sec

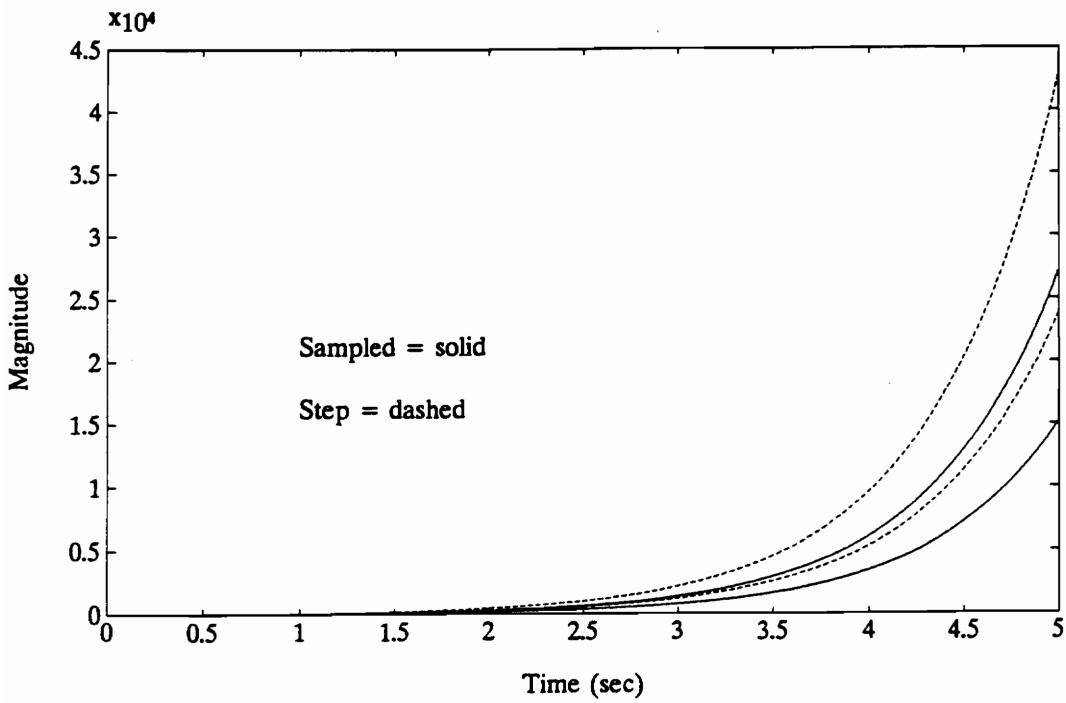

Figure 13. Outputs y(t) of Sampled and Step-Invariant Continuous-Time Models; $\Delta t = 0.5$ sec

4.3 Realistic Model

The larger order model presented in this section is a random, stable system which is more representative of a realistic system. The eigenvalues of the system matrix A were chosen randomly and are

$$\lambda(A)=\{-7.75, -5.15, -3.347, -2.03, -1.516, -4.825 \pm 1.452\,i, -1.379 \pm 0.917\,i\} \quad (4.3.1)$$

Matrices B and C are (9 x 3) and (5 x 9) pseudo-random matrices, respectively, with a standard deviation of one and a mean value of zero. Matrix D was chosen to be zero since few actual systems have a feedthrough term. These matrices correspond to the state-space representation defined by

$$\dot{x}(t) = A\ x(t) + B\ u(t) \quad (4.3.2)$$

$$y(t) = C\ x(t) + D\ u(t) \quad (4.3.3)$$

The input to this system is shown in Figure 14. The sampling time was chosen to be 0.1 seconds.

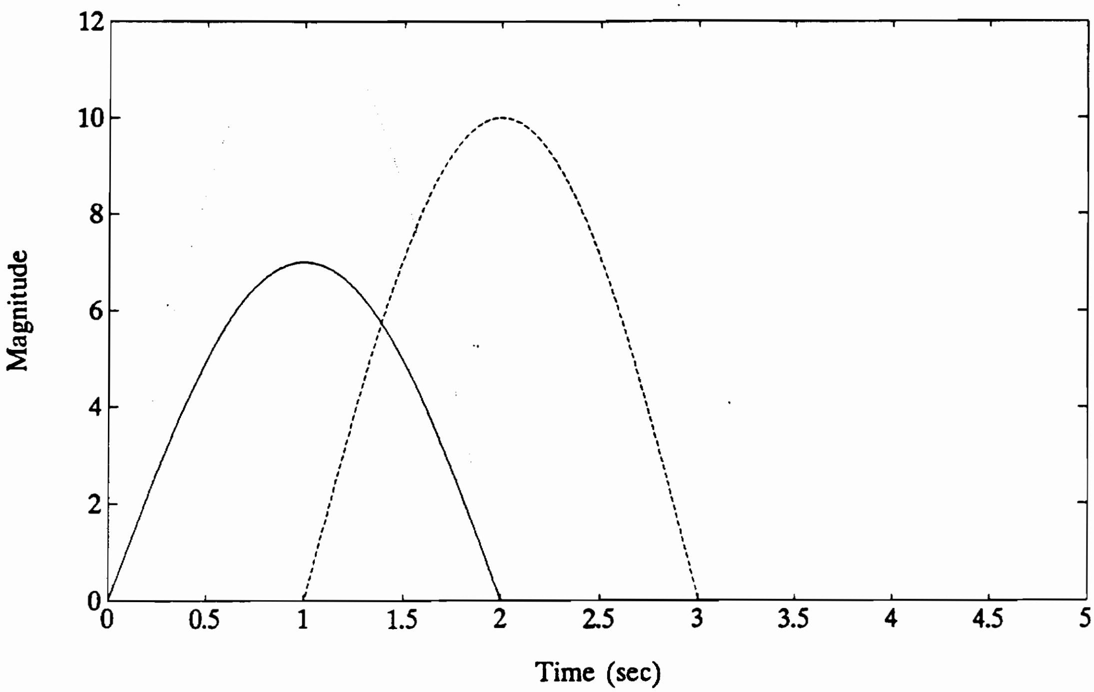

Figure 14. Input Signal u(t)

From the identified discrete-time model, $\{F_o, G_o, C_o, H_o\}$, corresponding ramp-invariant and step-invariant continuous-time representations were derived using the set pseudo-observability indices $\eta_i = \{2, 2, 2, 2, 1\}$. The continuous-time system matrix A_c was calculated as having the eigenvalues given below.

$$\lambda(A_c)=\{-8.04, -7.78, -5.052, -3.207, -1.537, -4.333 \pm 1.446i, -1.378 \pm 0.917i\} \quad (4.3.4)$$

The remaining matrices, namely B_f, C_f, D_f, B_z, C_z and D_z contain elements with relatively large magnitudes, therefore, they will not be given here. The elements of these matrices are really not as important as the response of the entire system.

Figures 15 and 16 show, respectively, the response of the ramp-invariant system and step-invariant system, each plotted with the sampled system response.

Since the error in the response of each output of the ramp- and step-invariant systems are the same, Figure 17 shows the error of the first output, $y_1(t)$, of each invariant system as compared with the sampled system. Note that although the magnitude of the error of the step-invariant representation is not excessively large in itself, it is large by comparison to the magnitude of the response of the sampled system.

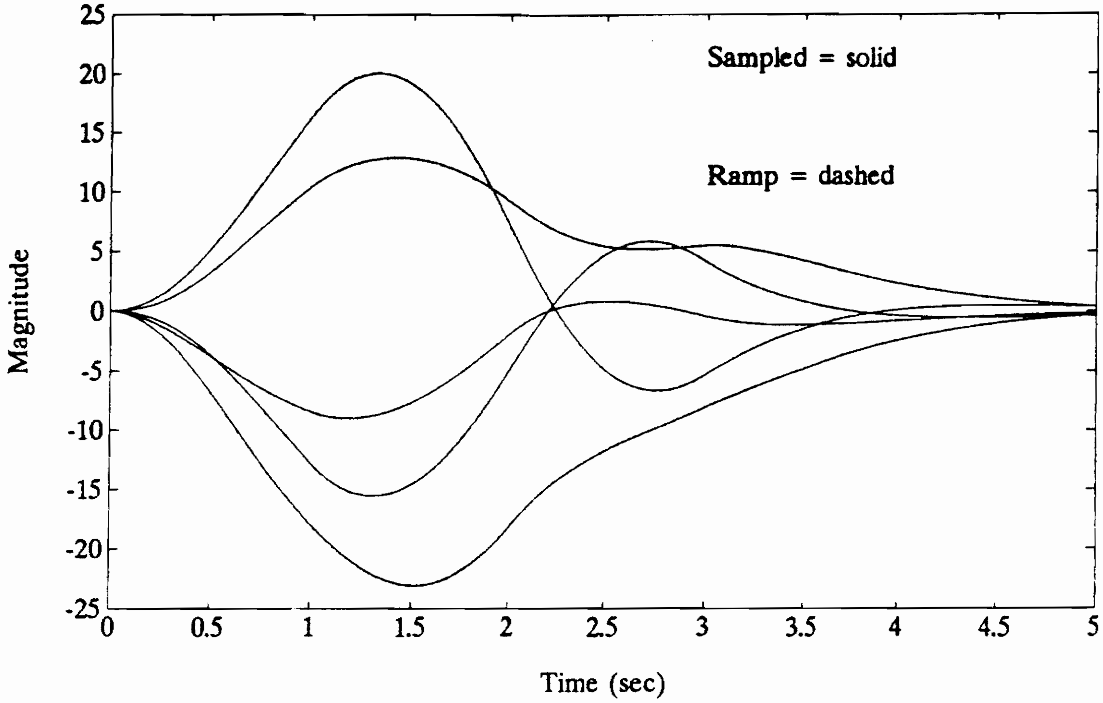

Figure 15. Outputs y(t) of Sampled and Ramp-Invariant Continuous-Time Models; $\Delta t = 0.1$ sec

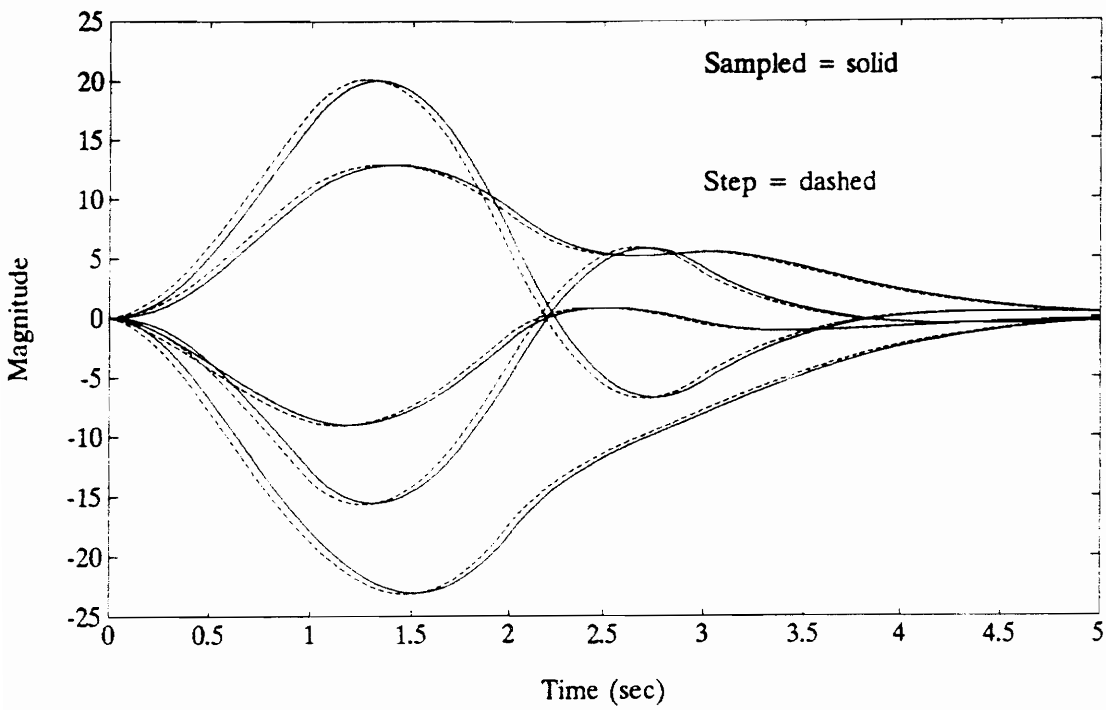

Figure 16. Outputs y(t) of Sampled and Step-Invariant Continuous-Time Models; $\Delta t = 0.1$ sec

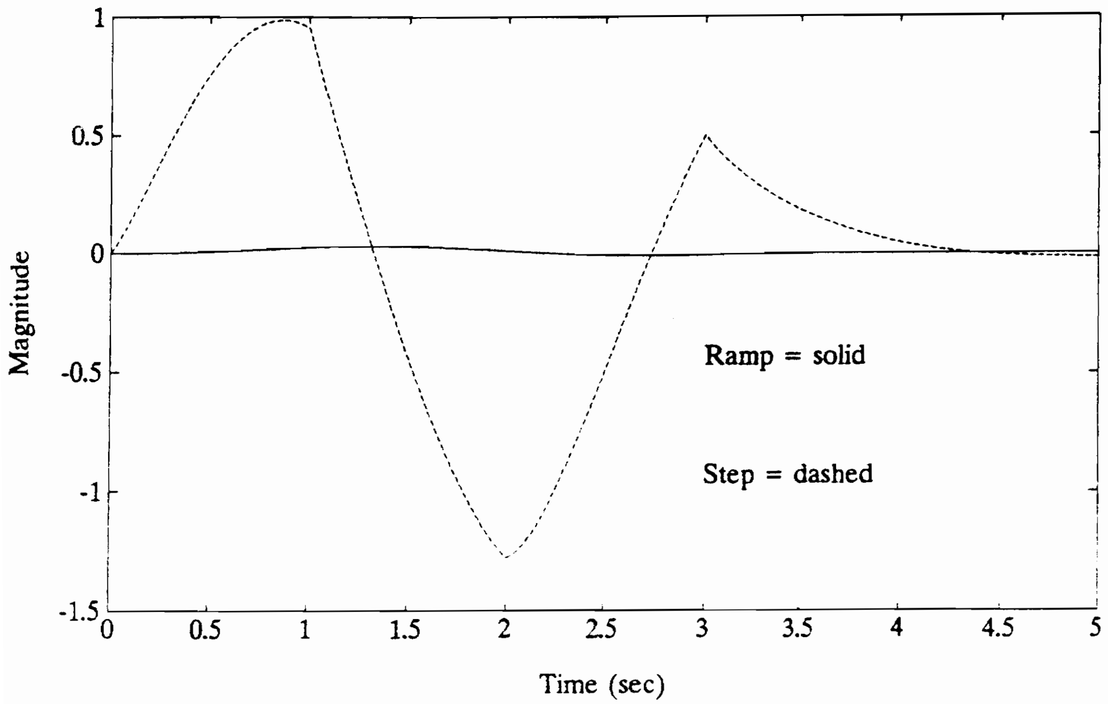

Figure 17. Differences between Sampled and Step-Invariant Models and Sampled and Ramp-Invariant Models.

4.4 Conclusions

These illustrative examples show that the techniques presented in chapters 2 and 3 are accurate and practical to use to identify a continuous-time representation of a given system. These methods, as presented, remove some of the previously posed restrictions in continuous-time identification. The examples in section 4.1 show a number of results. First, this section shows that the entire identification and transformation procedure works well for zero-valued eigenvalues and for multiple eigenvalues. Also, the fact that the ramp-invariant system should yield a better approximation to the original system than the step-invariant model, in general, is show to be true in these examples.

The main purpose behind section 4.2 is to show that the method of calculating the log of a square matrix which was given in Chapter 3.3 will work as stated. The unstable system and large sampling time used to produce the presented results were somewhat unrealistic but served the purpose of illustration. From a practical standpoint, this method could be employed to identify a model of an unstable system by "gently stimulating" the system to sample input-output data, which could be used to identify a model, which could in turn be used to design a stablizing compensator. Again, the ramp-invariant model is shown to be a better approximation to the original system than the step-invariant model.

The model in section 4.3 is more simple in some respects than the models used in the previous sections, in that this model has no zero eigenvalues and no unstable eigenvalues. The difficulty presented by this system is presented by the number of inputs and outputs and the order of the system. The models used in the previous sections were unintentionally SIMO models, which are by design, in most cases, easily identified. The true MIMO system presented in this section was identified easily and the responses of the identified models were both fairly accurate. Once again though, the ramp-invariant model was more accurate than the step-invariant model.

5.0 Conclusions

Numerous continuous-time multivariable system identification methods have been presented here. Direct identification methods were discussed briefly. Indirect identification methods were of primary consideration and some recently developed techniques were presented together for use as a method of identification.

The indirect identification method requires that a discrete-time system model be identified first. Many methods to accomplish this task have been presented to date. The method which was presented in this text was chosen for its accuracy and ease of use. The use of pseudo-observability indices by this method remove the need for structural identification separately from the identification of the system parameters. This is a major advantage over most of the previously introduced identification methods which were considered. Admissibility of the set of pseudo-observability indices used has been defined and presented.

The second step in the indirect identification of continuous-time systems is the transformation of the discrete model to an equivalent continuous-time representation. The classical Zero-Order hold (ZOH) method is considered and is relatively accurate for small sampling intervals. As the sampling interval increases, however, the accuracy of this method deteriorates. A new approach to the First-Order hold (FOH) method is presented. This method lacks the restrictions previously imposed on the FOH transformation which makes this method very attractive as it produces a more accurate system model than the ZOH method at any given sampling time.

Basic to both of these methods is the transformation of the discrete-time state transition matrix back to the continuous-time system matrix. This transformation is shown to be straightforward for "nice" system matrices but requires more stringent calculations when the system matrix is singular or the system is unstable. Again, a method is presented which removes the restrictions of previous methods and gives satisfactory results even in extreme cases.

Future work in this area would involve a Second-Order hold (SOH) or a higher order hold. With the speed of processors ever increasing, it could soon become feasible to calculate equivalent continuous-time representations of discrete-time models using higher order hold techniques. Another noteworthy point is that the identification method used to calculate the discrete-time model and the methods used to calculate the transformation of the discrete-time system to an equivalent continuous-time system, as presented in this thesis, can be used interchangeably with other compatable methods or transformations. This adds immense flexibility to one's toolbox of system identification techniques.

List of References

1. Bingulac, S.: 'CAD package L-A-S: A research and educational tool in systems and control', Proceedings of the 20th southeastern symposium on system theory, Charlotte, NC, 1988.

2. Bingulac, S.: 'Identification of multivariable dynamic systems', Proc. IFAC Symp. Ident. and Syst. Para. Esti., pp 1979-1987, 1978.

3. Bingulac, S.: 'On the minimal number of parameters in linear multivariable systems', IEEE Transactions on Automatic Control, vol. AC-21, pp. 604-605, 1976.

4. Bingulac, S. and Cooper, D. L.: 'Derivation of discrete- and continuous-time ramp invariant representation', Electronics Letters, vol. 26, no. 10, pp. 664-666, 1990.

5. Bingulac, S. and Djorovic, M.: 'On the minimality of the Jordan form state

equations', IEEE Transactions on Automatic Control, vol. AC-20, pp. 687-689, 1975.

6. Bingulac, S. and Farias, M. A. C.: 'Identification and minimal realization of multivariable systems', Proc. of the IFAC Symposium on Multivariable Technological Systems, Fredericton, Canada, pp. 95.1-95.5, 1977.

7. Bingulac, S. and Krtolica, R.: 'An algorithm for simultaneous order and parameter identification in multivariable systems', Proc. 8th IFAC Symposium on Identification and System Parameter Estimation, Beijing, China, pp. 542-547, 1988.

8. Bingulac, S. and Krtolica, R.: 'Generalized ARMA model for MIMO system identification', Proc. Automatic Controls Conference, Boston, pp. 1336-1341, 1985.

9. Bingulac, S. and Krtolica, R.: 'On admissibility of pseudoobservability indexes', IEEE Transactions on Automatic Control, vol. AC-32, pp. 920-922, 1987.

10. Bingulac, S., Padilla, R. A., and Padilla, C. S.: 'Modified canonical forms for linear multivariable systems', Automatica, vol. 22, pp. 83-88, 1981.

11. Bingulac, S. and Sinha, N. K.: 'On the identification of continuous-time multivariable systems from samples of input-output data', Proceedings of the 7th international conference on mathematical and computer modelling,

Chicago, 1989.

12. Brogan, W. L.: 'Modern control theory', Prentice-Hall, New Jersey, 1985.

13. Chen, C. F., and Hsiao, C. H.: 'A state-space approach to Walsh series solution of linear systems', International Journal of Systems Science, vol. 6, no. 9, pp 833-858, 1975.

14. Chen, C. F., Tsay, Y. T., and Wu, T. T.: 'Walsh operational matrices for fractional calculus and their applications to distributed systems', Journal of the Franklin Institute, vol. 303, pp. 267-284, 1977.

15. Chen, C. T.: 'Linear system theory and design', Holt, Rinehart, and Winston, Inc., New York, 1984.

16. Cooper, D. L., and Bingulac, S.: 'Computational improvement in the calculation of the natural log of a square matrix', Electronics Letters, vol. 26, no. 13, pp. 861-862, 1990.

17. Correa, G. and Glover, K.: 'On the choice of parametrization for identification', IEEE Trans. Auto. Control, vol. AC-31, no. 1, pp. 8-15, 1986.

18. Correa, G. O. and Glover, K.: 'Multivariable identification using pseudo-canonical forms', Proc. 6^{th} IFAC Symposium on Identification and System Parameter Estimation, Washington, D.C., pp. 1241-1246, 1982.

19. Denham, M. J.: 'Canonical forms for the identification of multivariable linear systems', IEEE Trans. Auto. Control, vol. AC-19, pp. 646-656, 1974.

20. Gevers, M. and Wertz, V.: 'Uniquely identifiable state-space and ARMA parametrizations for multivariable linear systems', Automatica, vol. 20, pp. 333-347, 1984.

21. Glover, K. and Willems, J. C.: 'Parametrizations of linear dynamical systems: canonical forms and identifiability', IEEE Trans. Auto. Control, vol. AC-19, pp. 640-645, 1974.

22. Gorti, B. M., Bingulac, S., and Vanlandingham, H. F.: 'Deterministic identification of linear multivariable systems', Proc. of the IEEE 22nd Southeast Symp. Syst. Theory, Cookville, TN, pp. 126-131, March, 1990.

23. Guidorzi, R.: 'Canonical structures in the identification of multivariable systems', Automatica, vol. 11, pp. 361-374, 1975.

24. Haykin, S. S.: 'A unified treatment of recursive digital filtering', IEEE Transactions on Automatic Control, vol. AC-17, pp. 113-116, 1972.

25. Hsia, T. C.: 'System identification', Lexington Books, Massachusetts, 1977.

26. Hsia, T. C.: 'On sampled data approach to parameter identification of continuous linear systems', IEEE Trans. Auto. Control, vol. AC-17, no. 2, pp. 247-249, 1972.

27. Hung, J. C., Liu, C. C., and Chou, P. Y.: 'An algebraic method for system parameter identification', Proc. 14^{th} Asilomar Conf. on Circuits, Systems, and Computers, Pacific Grove, California, pp. 277-279, 1980.

28. Kailath, T.: 'Linear systems', Prentice-Hall, New Jersey, 1980.

29. Keviczky, L.: 'On the equivalence of discrete and continuous transfer functions', Problems of Control and Information Theory, vol. 6, no. 2, pp. 111-128, 1977.

30. Keviczky, L. K. and Bányász, Cs. M.: 'Some new results on multiple input-multiple output identification methods', Proc. IFAC Symp. Ident. and Syst. Para. Est., pp. 1989-2001, 1978.

31. Lastman, G. J., Puthenpura, S. C., and Sinha, N. K.: 'Algorithm for the identification of continuous-time multivarible systems from their discrete-time models', Electronic Letters, vol. 20, no. 22, pp. 918-919, 1984.

32. Puthenpura, S. C. and Sinha, N. K.: 'Transformation of continuous-time model of a linear multivariable system from its discrete-time model', Electronic Letters, vol. 20, no. 18, pp. 737-738, 1984.

33. Sannuti, P.: 'Analysis and synthesis of dynamic systems via block-pulse functions', Proc. Inst. of Elec. Engrs., vol. 124, pp 569-571, 1977.

34. Sinha, N. K.: 'Estimation of transfer function of continuous system from sampled data', Proceedings IEE, vol. 19, pp. 612-614, 1972.

35. Sinha, N. K. and Kusta, B.: 'Modeling and identification of dynamic systems', Van Nostrand-Reinhold, New York, 1983.

36. Sinha, N. K. and Lastman, G. J.: 'Identification of continuous-time multivariable systems from sampled data', International Journal of Control, vol. 35, pp. 117-126, 1982.

37. Sinha, N. K. and Lastman, G. J.: 'Transformation algorithm for identification of continuous-time multivariable systems from discrete data', Electronics Letters, vol. 17, pp. 779-780, 1981.

38. Strmčnik, S. and Bremšak, F.: 'Some new transformation algorithms in the identification of continuous-time multivariable systems using discrete identification methods', Preprints, 5th IFAC Symposium on Identification and System Parameter Estimation, Darmstadt, Germany, pp. 397-405, 1979.

39. Strmčnik, S. and Bremšak, F.: 'An efficient algorithm for the transformation of the input-output model into the state-space model', International Journal of Control, vol. 41, no. 1, pp. 145-154, 1985.

40. Strmčnik, S. and Tramte, P.: 'On a generalized algorithm for the transformation of the input-output model into the state-space model', IEEE Transactions on Automatic Control, vol. 33, no. 5, pp. 472-474, 1988.

41. Suen, L. C., and Lui, R.: 'Determination of the structure of multivariable stochastic linear systems', IEEE Trans. Auto. Control, vol. AC-20, pp. 458-464, 1978.

42. Unbehauen, H. and Rao, G. P.: 'Identification of continuous systems', Elsevier Science Publishing Company, Inc, New York, 1987.

43. VanLandingham, H. F.: 'Introduction to digital control systems', Macmillan Publishing Co., New York, 1985.

44. Van Overbeek, A. J. M. and Ljung, L.: 'On-line structure selection for multivariable state-space models', Automatica, vol. 18, pp. 529-543, 1982.

45. Wertz, V., Gevers, M., and Hannan, E. J.: 'The determination of optimum structures for the state space representation of multivariable stochastic processes', IEEE Trans. Auto. Control, vol. AC-27, pp. 1200-1211, 1982.

46. Wolovich, W. A. and Guidorzi, R.: 'A general algorithm for determining state-space representations', pp. 295-299, 1976.

47. Zhou Qi-Jie and Sinha, N. K.: 'Identification of continuous-time state-space model from samples of input-output data', Electronics Letters, vol. 18, pp. 50-51, 1982.

Appendix

This appendix contains listings of the LAS subroutines which were used to perform the simulations presented in Chapter 4. The two subroutines given by Listings 8 - 12 and 13 - 14, respectively, perform identification using pseudo-observability indices. The subroutine IDMV.SBR, along with the subroutines which are called by it, form a subroutine for identification which is very thorough but time and memory consuming. The subroutine POID.SBR, along with the subroutine it calls, make up a subroutine for identification which is faster and requires less memory, yet does not warn the user that the output may be incorrect. Some of these programs are packaged with the LAS software.

Listing 1. LAS subroutine to calculate the response of a continuous-time system.

```
A,B,C,D,U,dt(rcs,sbr)=Y,NT,TT
(nli)=
(nty)=
_
_RESPONSE_OF_CONTINUOUS_SYSTEM
_INPUT_SYSTEM_&_INPUT
_OUTPUT_IS_IN_ROW_VECTOR_FORM
_
0(inc)=ONE
 A(rdi)=n
 n,1(dzm)=Xo
```

```
 U(rdi)=NT
 NT,ONE(-)=NTM1
 NTM1,dt(s*)=TT
 Xo,NT,TT,A,B,U(ce3)=XX
 C,XX(t)(*)=CX
 D(rdi)=dr
 D(cdi)=dc
 dr,1(ifj)=a,a,c
a:dc,1(ifj)=b,b,c
b:U(t),D(s*)=DU
 (jmp)=d
c:U(t)=UT
 D,UT(*)=DU
d:CX,DU(+)=Y
 Y(t)=Y
 (typ)=
 (lis)=
```

Listing 2. LAS subroutine to calculate the response of a discrete-time system.

```
A,B,C,D,U(rds,sbr)=Y
(nli)=
(nty)=
_
_RESPONSE_OF_DISCRETE_SYSTEM
_USED_FOR_SYSTEMS_WITH_A_FEED-THROUGH_MATRIX_D
_INPUT_SYSTEM_&_INPUT
_OUTPUT_IS_IN_COLUMN_VECTOR_FORM
_
 U(rdi)=NT
 U(mcp)=UU
 NT(dec)=NT
 A(rdi)=n
```

```
  B(cdi)=m
  C(rdi)=p
  n,1(dzm)=X
  X(mcp)=X2
  1(dec)=j
a:j(inc)=j
  A,X(*)=AX
  UU,1(ctr)=uu,UU
  B,uu(s*)=BU
  AX,BU(+)=X1
  X2,X1(cti)=X2
  X1(mcp)=X
  j,NT(ifj)=a,b,b
b:(nop)=
  U,1(ctr)=XXX
  C,X2(*)=CX
  D(rdi)=dr
  D(cdi)=dc
  dr,1(ifj)=c,c,e
c:dc,1(ifj)=d,d,e
d:U(t),D(s*)=DU
  (jmp)=f
e:U(t)=UT
  D,UT(*)=DU
f:CX,DU(+)=Y
  Y(t)=Y
  (typ)=
  (lis)=
```

Listing 3. LAS subroutine to sample input-output data for identification.

```
Y,DT1,DT2(sam2,sbr)=YNEW
(nli)=
```

```
(nty)=
Y(rdi)=NT
NT(dec)=NTM1
Y,NTM1(ctr)=Y,YBOT
Y(cdi)=p
Y(t)=YY
DT2,DT1(s/)=SATI
 NTM1,SATI(s/)=COL
 COL,0(dzm)=YNEW
 1(dec)=j
a:j(inc)=j
 YY,1(ctr)=Y1,YY
 Y1,COL(vtm,t)=MAT
 MAT,1(ctr)=Y1T
 YNEW,Y1T(t)(cti)=YNEW
 j,p(ifj)=a,b,b
b:(nop)=
 YNEW,YBOT(rti)=YNEW
 (typ)=
 (lis)=
```

Listing 4. LAS subroutine to calculate the logarithm of a square matrix.

```
F,dt,eps(logm,sbr)=Ac,JJ
(nli)=
(nty)=
_LOGM
_PROGRAM_TO_ESTIMATE_A=1/T*ln(F)
_USING_ALGORITHM_FROM_LASTMAN_PUTHENPURA_SINHA
0(inc)=MAX
MAX,2(s/)=MIN
MAX,100(s*)=MAX
 F(rdi)=N
```

```
  N,N(dim)=I
  1(dec)=j
a:j(inc)=j
  F,I(-)=L
  L(egv)(rpt),1(ctc)=EGL,EGl
  EGL(max)=SPRD
  SPRD,MIN(ifj)=c,c,b
b:F(sqm)=F
  (jmp)=a
c:1(dec)=K
  L,L(mcp)=S,Q
  j(dec)=j
  j(mcp)=JJ
d:K(inc)=K
  L,Q(*),-1(s*),K(s*)=QNUM
  K(inc)=QDEN
  QNUM,QDEN(s/)=Q
  S,Q(+)=SK+1
  SK+1,S(-)(abs)(max)=DNUM
  SK+1(abs)(max)=DDEN
  DNUM,DDEN(s/)=D
  K,MAX(ifj)=e,g,g
e:D,eps(ifj)=g,f,f
f:SK+1(mcp)=S
  (jmp)=d
g:SK+1,dt(s/)=AX
h:j,0(ifj)=l,l,k
k:AX,2(s*)=AX
  j(dec)=j
  (jmp)=h
l:AX(mcp)=Ac
  (typ)=
  (lis)=
```

Listing 5. LAS subroutine to perform Zero-order Hold, or step-invariant transformation of a discrete-time system to a continuous-time system.

```
Ad,Bd,Cd,Dd,Ac(zoh,sbr)=Bz,Cz,Dz
(nli)=
Ac(rdi)=n
n,n(dim)=I
Ad,I(-)=AoI
AoI(-1)=AoIi
Ac,AoIi,Bd(*)=Bz
Cd(mcp)=Cz
Dd(mcp)=Dz
(lis)=
```

Listing 6. LAS subroutine to perform a First-order Hold, or ramp-invariant transformation of a discrete-time system to a continuous time system.

```
Ai,Bi,Ci,Di,Ac,dt(rit2,sbr)=Bc,Dc,Mo,M1
(nli)=
(nty)=
_RIT2
_RAMP_INVARIANT_TRANSFORMATION
_USING_POLYNOMIAL_MATRICES
0(inc),20(s*)=tt
Ai(rdi)=n
Bi(cdi)=m
Ci(rdi)=p
n,n(dim)=In
p,p(dim)=Ip
dt,Ac,tt(foh,sub)=Mo,M1,rrr
```

```
  (nli)=
x:M1,Mo(-1)(*)=P
  Ai,Bi,Ci(mtf)=ch,W
  Ai,In,Ci(mtf)=ch,V
  0,m(dzm)=Gm
  0,n(dzm)=Gn
  0,p(dzm)=ChI
  0,m(dzm)=ChDi
  1(dec)=i
i:i(inc)=i
  ch,1(ctc)=chi,ch
  W,1(ctr)=wi,W
  V,1(ctr)=vi,V
  Gm,wi,m(vtm)(rti)=Gm
  Gn,vi,n(vtm)(rti)=Gn
  ChI,Ip,chi(s*)(rti)=ChI
  ChDi,Di,chi(s*)(rti)=ChDi
  i,n(ifj)=i,j,j
j:(nop)=
  p,m(dzm)=pmz
  p,n(dzm)=pnz
  Gm,pmz(rti)=Gm
  ChDi,Di(rti)=ChDi
  Gm,ChDi(+)=GmCh
  Gn,pnz(rti)=Gn1
  pnz,Gn(rti),P(*)=Gn2
  Gn1,Gn2(+),ChI,Ip(rti)(cti)=GnCh
  GnCh,GmCh(sle),n(ctr)=Go,Dc
  Mo(-1),Go(*),dt(s/)=Bc
  (typ)=
  (lis)=
```

Listing 7. LAS subroutine called by RIT2 to calculate matrices M_0 and M_1

```
 dt,Acc,im(foh,sub)=Mbo,Mb1,Mb
 (nli)=
 1(dec)=i
 i(inc)=one
 one(inc)=two
 one,100(s/)=eps
 Acc(cdi)=n
 n,n(dim)=In
 Acc(svd)=w
  w,w,eps(f/,t)=x
  x,x(t)(*,t)=rank
  (jmp)=s
  n,rank(ifj)=r,r,s
r:(nop)=
  Acc,dt(s*)=Act
  Act(-1)=Acti
  Acti,Acti(*)=Act2
  dt,Acc(eat)=Ad
  Act,Ad(*),Ad(-),In(+),Act2(*,t)=Mbon
  Ad,Act(-),In(-),Act2(*,t)=Mb1n
  Acti,Ad,In(-)(*,t)=Mbn
  Mbn,Mb1n,Mbon(mcp)=Mb,Mb1,Mbo
  (jmp)=j
s:(nop)=
  In(mcp)=Mb
  In,2(s/)=Mbo
  Mbo(mcp)=Mb1
  one(mcp)=fac1
  two(mcp)=fact
```

```
  Acc,dt(s*)=Acdt
  Acdt(mcp)=Acti
i:i(inc)=i
  fact,fac1,i(cti)=fi
  i(inc)=i1
  i1(inc)=i2
  i2,fact(s*,t)=fact
  fac1,i1(*)=fac1
  Mb,Acti,fac1(s/)(+,t)=Mb
  Mbo,Acti,i1,fact(s/)(s*)(+,t)=Mbo
  Mb1,Acti,fact(s/)(+,t)=Mb1
  Acti,Acdt(*)=Acti
  i,im(ifj)=i,j,j
j:(nop)=
  (typ)=
  (lis)=
```

Listing 8. LAS subroutine to identify a state-space representation of a system in pseudo-observable canonical form.

```
u,y,nmp,nv(idmv,sbr)=Ao,Bo,Co,Do,xo,cond
(nli)=
(nty)=
_Samples_{ui,yi}_==>_Discrete_Time_Model
_{Ao,Bo,Co,Do,xo}_in_Pseudo_Observable_Form
_L-A-S_Subroutine_called:
_____VNMI,sub__,QC1,sub__,BCU1,sub__and__,__H1.sub
nmp(tvc,t)=n,m,p
nv(poi)=n,nx,va,vi,v1,v2
 u,nx(vnmi,sub)=U
 (nli)=
 U,u(cti),nx(ctr)=x,U
 (elm)=u
```

```
U(svd,t)=wu
y,nx(vnmi,sub)=Y
(nli)=
Y,y(cti),nx(ctr)=x,Y
(elm)=y
Y,v1(dsm)(*)(t)=Y1
Y1(t)(svd,e)=wy
Y,v2(dsm)(*)(t)=Y2
U(t),Y1(rti)=Z
Z(rdi),Z(cdi)(mcp)=nrZ,ncZ
Z(t)(svd)=wz
wz,1,wz(cdi),1,1(exm),wz,1(ctc)(s/,e)=cond
nrZ,ncZ,cond(cti)=dcnd
dcnd(out,e)=
Z(t),Y2(t)(sle)(t,t)=BA
BA,nx(inc),m(*)(ctc,t)=Bt,At
n,n(dim),p(ctr)=Co,A2
va(dsm),At(*),vi(dsm),A2(*)(+,t)=Ao
Ao(egv)(t,t)=eid
va(dsm,t)=Be
Ao,Be(qc1,sub)=Qc
(nli)=
Bt,m(bcu1,sub)=Btt
(nli)=
Qc,Btt(rdi)(ctc),Btt(*,t)=Bo
Bt,m(ctc)=Bt1
At,p(ctc)(-1),Bt1(*),Co,Ao(-1),Bo(*)(*)(-,t)=Do
Do,-1(s*)=Do
Ao,Bo,Co(mtf,t)=fo,Go
n,p(-)(inc)(inc),p(*,t)=np1p
Ao,Bo,Co,Do,np1p,nx(h1,sub)=SS
(nli)=
Y1,1(ctc),U(t),1(ctc)(mcp)=Y11,U1
SS,v1(cdi)(ctr)=SS
Y11,v1(dsm)(t),SS(*),U1,nx,m(*)(ctr)(*)(-,t)=xo
```

```
  (typ)=
  (lis)=
```

Listing 9. LAS subroutine called by IDMV.SBR.

```
  v,n(vnmi,sub)=M
  (nli)=
  v(shd)=v1
  v1(mcp)=M
  n(dec)=i
i:v1(shd)=v1
  v1,M(cti,t)=M
  i(dec,t)=i
  i(ifj)=j,j,i
j:(lis)=
```

Listing 10. LAS subroutine called IDMV.SBR.

```
  a,b(qc1,sub)=qc
  (nli)=
  b(cdi)=m
  a(rdi)=n
  b,b(mcp)=qc,x
n:n(dec)=n
  a,x(*)=x
  qc,x(cti,t)=qc
  n,m(ifj)=f,n,n
f:(lis)=
```

Listing 11. LAS subroutine called by IDMV.SBR.

```
  Ba,m(bcu1,sub)=b123
  (nli)=
  Ba(mcp)=x
  0,m(dzm)=b123
i:x,m(ctc)=y,x
  b123,y(rti,t)=b123
  x(cdi)(ifj)=j,j,i
j:(lis)=
```

Listing 12. LAS subroutine called by IDMV.SBR.

```
  Aid,Bid,Cid,Did,np1p,nimx(h1,sub)=SS
  (nli)=
  Aid(cdi),Bid(cdi),Cid(rdi)(mcp)=n,m,p
  np1p,m,nimx(*)(dzm)=SS
  n,n(-)=j
  nimx(inc)=im
  n,n(dim)=Ai
J:j(inc)=j
  n,n(-)(inc)=jj
    j(dec),p(*)(inc)=ii
    im(dec)=im
    j,1(ifj)=C,o,C
o:Did(mcp)=AjBD
    (jmp)=R
C:(nop)=
    Cid,Ai,Bid,m(ctc)(*)(*)=AjBD
    Ai,Aid(*)=Ai
R:(nop)=
    n,n(-)=i
I:i(inc)=i
    SS,AjBD,ii,jj(rmp)=SS
    _SS(out)=
```

```
  _(sto)=
  jj,m(+)=jj
  ii,p(+)=ii
  i,im(ifj)=I,K,K
K:j,nimx(ifj)=J,L,L
L:(lis)=
```

Listing 13. LAS subroutine to identify a state-space representation of a system in pseudo-observable canonical form. More time efficient than Listing 8.

```
 u,y,poi(poid,sbr)=AO,BO,CO,DO
 (nli)=
 _SYSTEM_IDENTIFICATION
 _USING_PSEUDO_OBSERVABILITY
 _INDICES
 poi(poi)=n,nm,be,bec,s,sz
 be(dsm)=be
 bec(dsm)=bec
 s(dsm)(t)=s
  sz(dsm)(t)=sz
  u(cdi)=m
  u(rdi)=ur
  y(cdi)=p
j:nm(dec)=nm-1
  nm(inc)=nm+1
  ur,nm(-)=q
  u,y(T)=uo,yo
  uo,yo(mcp)=uu,yy
  -1(inc)=i
a:i(inc)=i
  yy(shl)=yy
  yo,yy(rti)=yo
```

```
  i,nm(ifj)=a,b,b
b:-1(inc)=j
  (elm)=y,yy
c:j(inc)=j
  uu(shl)=uu
  uo,uu(rti)=uo
  j,nm(ifj)=c,d,d
d:s,yo(*)=y1
  (elm)=u,uu
  sz,yo(*),q(ctc)=y2,e
  uo,y1(rti),q(ctc)=z,e
  z(t),y2(t)(sle)(t)=ba
  nm(inc),m(*)=mnm
  ba,mnm(ctc)=by,ay
  by,m(ctc)=by1
  n,n(dim)=I
  I,p(ctr)=e,I2
  bec,I2(*)=bei
  be,ay(*),bei(+)=AO
  by(mcp)=yb
  0,m(dzm)=bk
  -1(inc)=k
e:k(inc)=k
  yb,m(ctc)=kb,yb
  bk,kb(rti)=bk
  k,mnm(ifj)=e,f,f
f:(nop)=
  AO,be,mnm(qc2,sub)=qc
  qc,bk(*)=BO
  p,n(dim)=CO
  ay,p(ctc)(-1),by1(*),CO,AO(-1),BO(*)(*)(-)=DO
  DO,-1(s*)=DO
  (lis)=
```

Listing 14. LAS subroutine called by POID.SBR.

```
  a,b,r(qc2,sub)=qc
  (nli)=
  a(rdi)=n
  r(dec)=rr
  b,b(mcp)=qc,x
  -1(inc)=i
n:i(inc)=i
  a,x(*)=x
  qc,x(cti,t)=qc
   i,rr(ifj)=n,f,f
 f:(nop)=
```

Vita

David L. Cooper, II was born September 11, 1965 in Camilla Georgia where he grew up and graduated from Westwood High School in June, 1983. After attending Albany Junior College for one year, he enrolled at the University of Georgia where he obtained a B.S. in Agricultural Engineering in June, 1988.

He began graduate school at the Virginia Polytechnic Institute and State University in August, 1988 and finished in September 1990, receiving the Master of Science in Electrical Engineering.

David L. Cooper, II