ORDER REDUCTION OF LINEAR SYSTEMS

BY PÄDE APPROXIMATION METHODS

Yacoub Shamash

A thesis submitted for the
degree of Doctor of Philosophy

Computing and Control Department
Imperial College of Science and Technology
University of London

August 1973
ABSTRACT

In this thesis methods are developed for the reduction in order of linear systems. The methods are based on the theory of Padé approximation.

A method of reduction, based on expanding the high order system transfer function into a J-type continued fraction, is introduced. The reduced models derived by this method are equivalent to the corresponding models derived by C-type continued fractions and by the time-moments method. The methods are special cases of Padé approximation.

Biased reduced models, in the sense that the model approximates the initial transient response of the system more closely than the steady state response and vice-versa, are derived using continued fraction expansions.

The above methods are extended to the reduction of multivariable systems, and hence it is necessary to invert matrix transfer functions. An algorithm for inverting linear time-invariant multivariable systems, based on the theory of Padé approximations, is introduced.

Minimal realization algorithms are used to derive reduced order models. Reduction is effected by the simple partitioning of some system matrices. An algorithm for the minimal realization of systems described by a set of constant differential equations is introduced. A minimal realization of a system may be derived by using a combination of its Markov parameters and time-moments.

A new method of reduction, based on the retention of some modes of the high order system in the model and on the concept of Padé approximation about more than one point, is introduced. The method
retains the desirable features of the modal and Padé approximation methods. It has the important property that for a stable system the reduced model is always stable.

A procedure for ensuring the stability of reduced models derived by Padé approximations is given.

Finally the methods of the thesis are extended to the problem of reducing discrete-time systems.
Acknowledgements

I would like to thank sincerely my supervisor Dr. J.M.C. Clark for his guidance on the research work leading to this thesis.

I would also like to thank Dr. J.C. Allwright and Mr. H. Johnson for their useful discussions. I would also like to thank my fellow students, among them B.M.N. Clarke, D.B.H. Costanos, J. Katzberg and M. Denham for their interest and comments.

This work was made possible by the provision of a University of London Postgraduate Studentship.

Finally, my thanks must go to Mrs. L. Rice for her efficient typing of this thesis.
CONTENTS

Abstract page 2
Acknowledgements 4

Chapter 1 INTRODUCTION TO THE THESIS
1.1 Introduction 9
1.2 Contributions of the thesis 13

Chapter 2 A SURVEY OF MODEL REDUCTION TECHNIQUES AND THEIR APPLICATIONS
2.1 Introduction 15
2.2 Applications of model reduction 17
2.3 Survey of model reduction techniques 20
2.3.1 Modal methods 21
2.3.2 Reduction by minimisation of an error criterion 25
2.3.3 Reduction in the time-domain 27
2.3.4 Reduction in the frequency-domain 31
2.3.5 Analytic methods of reduction 35
2.3.6 Miscellaneous methods of reduction 38
2.4 Discussion and comments 41

Chapter 3 CONTINUED FRACTION METHODS FOR THE REDUCTION OF SINGLE INPUT-SINGLE OUTPUT SYSTEMS
3.1 Introduction 43
3.2 Chen and Shieh's method of reduction 45
3.3 Model reduction using J-type continued fractions about s = 0 48
3.3.1 J-type expansions for systems described in state-vector form 52
3.3.2 Reduced order models with fixed numerator orders 53
3.4 Padé approximation about s = 0 and s = \infty 54
3.4.1 Comparison of Padé approximation techniques with the continued fraction methods 58
3.5 Biased reduced models using continued fractions 58
3.6  Justifications for using Padé type approximations  64
3.7  Examples  69

Chapter 4  CONTINUED FRACTION METHODS FOR THE REDUCTION OF
CONSTANT LINEAR MULTIVARIABLE SYSTEMS

4.1  Introduction  72
4.2  Padé approximation of analytic matrices  74
4.3  Model reduction using continued fraction expansions about s = 0  76
4.3.1  Equivalence of the two methods  79
4.3.2  Continued fraction expansion for systems described by their
time-moments  81
4.4  Model reduction using biased continued fraction expansion  83
4.5  Padé approximation of multivariable systems  85
4.5.1  Padé approximation about s = 0  85
4.5.2  Padé approximation about s = 0 and s = ∞  88
4.5.3  Comparison of Padé approximation and continued fraction methods  89
4.6  Example 1  90
4.7  Padé approximation for the reduction of a general multivariable
system  93

Chapter 5  CONSTRUCTION OF THE INVERSE OF LINEAR TIME-INVARIANT MULTIVARIABLE SYSTEMS

5.1  Introduction  100
5.2  Construction of the system transfer function from its power
series expansion  101
5.3  To compute the transfer function of a system described in
state-space form  105
5.4  Criterion for the existence of the inverse  109
5.4.1  L-integral inverses  109
5.4.2  Test for invertibility  110
5.5  Inversion algorithm  111
5.5.1  The first non-zero Markov parameter is non-singular  111
5.5.2 The first non-zero time-moment is non-singular
5.5.3 The general case
5.6 Generalised inverse

Chapter 6 MINIMAL REALIZATION THEORY AND THE MODEL REDUCTION PROBLEM

6.1 Introduction
6.2 Minimal realization of linear dynamical systems
6.2.1 Realizability in terms of Markov parameters
6.2.2 Realizability in terms of time-moments
6.2.3 Realizability in terms of time-moments and Markov parameters
6.3 Realization of a set of constant differential equations
6.3.1 Expansion of the system to get the Markov parameters
6.3.2 Expansion of the system to get the time-moments
6.3.3 The general case
6.4 Minimal partial realization and its relation to the reduction problem
6.5 Application of minimal realization algorithms to the reduction of linear systems
6.5.1 State-vector representation of reduced order models derived by Padé approximation about \( s = 0 \)
6.5.2 State-vector representation of biased reduced order models
6.5.3 Reduction of multivariable systems

Chapter 7 LINEAR SYSTEM REDUCTION USING PADÉ APPROXIMATION TO ALLOW RETENTION OF DOMINANT MODES

7.1 Introduction
7.2 Evaluation of polynomial zeros using Koenig's theorem
7.3 Padé approximation and dominant mode reduction
7.3.1 An alternative interpretation using partial fractions
7.3.2 Examples
7.4 Reduction of multivariable systems
7.5 Biased reduced order models 169
7.6 Advantages and disadvantages of the method 172
7.7 The derivation of stable reduced order models using Padé approximation 177

Chapter 8 THE REDUCTION OF DISCRETE-TIME SYSTEMS
8.1 Introduction 185
8.2 Definition of time-moments and Markov parameters 185
8.3 Continued fraction expansion about z = 1 187
8.4 Continued fraction expansion about z = 1 and z = ∞ 190
8.5 Padé approximation to allow retention of dominant modes 198

Chapter 9 SUMMARY AND CONCLUSION 200

References 204
Appendix I PROPERTIES OF CONTINUED FRACTIONS 211
Appendix II PROPERTIES OF PADE APPROXIMATION 216
Chapter 1

INTRODUCTION TO THE THESIS

1.1 Introduction

A number of different methods have been proposed over the past seven years for the reduction of high order dynamic systems. All the methods have the same objective: to provide a low order linear model of the complex system, which is computationally or analytically more tractable than the "complete" system model, yet still provide sufficient information about the system to be useful. Many of the methods, however, are limited by the numerical computations required to derive the reduced order model. The most common applications of reduced order models are in simulation and control.

In Chapter 2, the problem of model reduction is formulated and some desirable features of reduced models stated. Applications of model reduction techniques in modern control technology are outlined. A survey of model reduction techniques is given and a critical analysis of several methods is made.

As will be seen from Chapter 2, a large number of techniques have been developed for the reduction of systems. Some are applicable to systems described by a transfer function, while others are applicable to systems described in state-vector form, and others are applicable to systems described by input/output measurements. Some of the methods are simple, while others are very complex.

Our purpose has been to derive a method of reduction which is conceptually and computationally simple and is of general applicability, that is it may be applied to systems described by a transfer function, in state-vector form, or by a set of input/output measurements.

It has been recognised [2, 14] that one of the most powerful methods of reduction is that by Chen and Shieh [19], which is based on
expanding the higher order system transfer function into a continued fraction (C-type). Reduced order models are then derived by truncating the continued fraction and inverting. The method is computationally very simple, and in many studies [91, 96] have been found to produce very good approximations. Thus we have taken Chen and Shieh's method as the starting point.

In Chapter 3, the method of Chen and Shieh is shown to be equivalent, in the case of asymptotically stable systems, to the method of reduction based on equating the initial time-moments of the system and model [8]. The two methods are special cases of the Padé approximation technique, which is widely used in problems in physics and chemistry [11, 85]. A new continued fraction expansion (J-type) is introduced which is computationally more efficient than the C-type continued fraction. Padé approximation is used to derive equivalent reduced order models and the advantages and disadvantages of the procedure in comparison with the continued fraction methods are discussed. Biased reduced order models, in the sense that the model is made to approximate the initial transient response of the system more closely than the steady state response and vice versa, are derived using continued fraction expansions about \( s = 0 \) and \( s = \infty \). The methods may be applied to systems described by a state-vector formulation directly without having to compute the transfer function of the system. Some physical justifications for using this type of approximation are given.

In Chapter 4, the methods of Chapter 3 are extended to the problem of reducing the order of constant multivariable systems. J-type continued fraction expansions are used to derive reduced order models of equal input-equal output systems. The method, although giving the same reduced models as Chen's method [22], is computationally superior. Further, biased reduced order models are derived using continued
fraction expansions about \( s = 0 \) and \( s = \infty \). Pade\' approximations, as defined by Bessis [11], about \( s = 0 \), and \( s = 0 \) and \( s = \infty \) respectively are used to derive reduced models which, when they exist, are the same as those derived by continued fraction methods. Pade\' approximation is extended to the problem of reducing general multivariable systems.

The use of continued fractions for the reduction of multivariable systems may require the inversion of rational transfer function matrices, while the use of Pade\' approximation would necessitate the inversion of a polynomial matrix. Hence in Chapter 5, an algorithm for the inversion of systems based on the theory of Pade\' approximation, is introduced. It is equally applicable to systems described by a matrix transfer function or a state-vector representation. Further, a method is introduced for computing the transfer function of a system described in state-vector form.

In Chapter 6, the problem of minimal realization of linear time-invariant systems is described and it is shown how a system may be realized from its Markov parameters, time-moments, or a combination of both. An algorithm is introduced for the minimal realization of systems described by a set of constant differential equations, which is based on existing algorithms for the minimal realization of systems described by input/output measurements [1, 43, 77, 89]. This is useful if a state-vector representation of the reduced models derived in Chapter 4 is required. The problem of minimal partial realization is described and its relation to the model reduction problem established. It is then shown how existing algorithms for the minimal realization problem may easily be applied to derive reduced order models which are equivalent to those of Chapter 3. The algorithms may also be used for the reduction of general multivariable systems. Hence if any of the algorithms [1, 43, 77, 89] is available, then it
is unnecessary to write a model reduction algorithm.

A serious shortcoming of the continued fraction and Padé approximation methods is that the reduced model may be unstable (stable) even though the high order system is stable (unstable). This is discussed in Chapter 7, and an algorithm based on the retention of one or more poles of the system in the reduced model is introduced for overcoming it. The algorithm will always lead to a stable model. In Chapter 7, a new method of reduction is introduced which combines the desirable features of the Padé approximation method and the modal methods [28, 60]. The reduced order model is constrained to retain the dominant poles (or any other desirable poles) of the high order system. The rest of the coefficients of the reduced model are chosen such that the model approximates the high order system in the Padé sense. The method is based on the use of Koenig's theorem and its generalisation for the convergence of poles of Padé approximants [44] to compute the dominant poles of the system in one operation, rather than computing individual poles of the high order system. The amount of computations involved in using this method is the same as that required for the Padé approximation. The method has the very important property of deriving reduced models which are stable (unstable) if the system is stable (unstable).

In Chapter 8, the methods of reduction discussed earlier are extended to the problem of reducing the order of linear time-invariant discrete-time systems.

Throughout the thesis, the different methods are illustrated by numerical examples. The adequacy of the reduced models have been tested by considering the step responses of the system and the model. The step response has been chosen, because it has been very widely used, for this purpose, in the literature.

Appendix I, contains some notations and basic definitions of
continued fractions, and Appendix II contains a summary of the theory of Padé approximations. It is recommended that the reader should familiarise himself with the contents of the Appendices before reading the thesis.

1.2 Contributions of the Thesis

Chapter 3

In this Chapter, the following are believed to be original:

Establishing the equivalence of Chen and Shieh's method of reduction, for asymptotically stable systems, and the method of time-moments.

The method of reduction based on expanding the system transfer function into a J-type continued fraction, and truncating it to get the reduced model.

The proof that the reduced models derived by C-type and J-type continued fractions are equivalent.

Deriving continued fraction expansions of systems described in state-vector form directly without computing the corresponding system transfer function.

The use of continued fraction methods for synthesising biased reduced order models.

A paper by the author based on the contents of this Chapter was presented at the Conference on Computer Aided Control Systems Design, held at Cambridge, April 1973 [86].

Chapter 4

In this Chapter the following are believed to be new:

The use of J-type continued fractions for the reduction of equal input-equal output multivariable systems.

The proof that the reduced models derived by C-type and J-type continued fractions are equivalent, and are a special case of Padé

The use of Padé approximation for the reduction of multivariable systems.

The use of Padé approximation about more than one point, and continued fraction methods to derive biased reduced models.

Chapter 5

Apart from the definitions and theorems of Section 5.4, results in this Chapter are original.

Chapter 6

In this Chapter, the following are believed to be original.

The minimal realization of a system from a combination of its Markov parameters and time-moments.

The algorithm for the minimal realization of a system described by a set of constant differential equations, using existing algorithms for the realization of systems based on the Hankel matrix approach.

The use of minimal realization algorithms for the reduction of systems in the sense of Padé approximation.

A paper by the author based on the contents of Sections 6.2 and 6.3 of this Chapter was presented at the International Conference on Systems Control, held in India, August 1973 [87].

Chapter 7

Apart from the statement of Koenig's theorem, and its generalisation for the convergence of poles of Padé approximants, the results and algorithms of this Chapter are original.

A paper by the author, based on the method of reduction introduced in Section 7.3, was read at the International Conference on Padé Approximations and their Applications, at Kent University, July 1972 [85].

Chapter 8

The results of this Chapter are original.
Chapter 2
A SURVEY OF MODEL REDUCTION TECHNIQUES
AND THEIR APPLICATIONS

2.1 Introduction

Model reduction is the process of reducing the mathematical complexity of a system model. In the case of linear time-invariant systems, reduction of complexity is synonymous with the reduction of dynamic order of the set of equations constituting the model. This is not necessarily the case with nonlinear or time-varying systems [63]. A second order system with complicated time variations is not necessarily simpler than a fourth order system with simple time variations. In this thesis, unless otherwise stated, we will be solely concerned with the reduction in order of linear time-invariant systems.

Model reduction techniques are of growing importance in modern control technology. In Section 2.2, some of the applications of model reduction techniques in control system design are given.

The problem of model reduction is a fairly recent topic, and research into the problem did not start in earnest until 1966-7, after the publication by Nicholson [68] of a method of controlling an oil-fired boiler by using a reduced model for generating suboptimal controllers. Since then, a number of methods have been developed for deriving reduced order models to approximate high order systems. In Section 2.3, a survey of some of the available methods of reduction is given.

For systems described in state-vector form, the model reduction problem may be formulated as follows:

Let the high order system be given by
\[ \dot{x} = Ax + Bu \quad (2.1.1) \]
\[ y = Cx \]
where \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^q \), \( y \in \mathbb{R}^p \) and \( A, B, C \) are constant matrices with dimensions compatible with the vectors \( x \), \( u \), and \( y \).

The problem is then to find a simplified model of a reduced order \( r < n \) given by
\[ \dot{x}^* = A^*x^* + B^*u \quad (2.1.2) \]
\[ y^* = C^*x^* \]
where \( x^* \in \mathbb{R}^r \), \( y^* \in \mathbb{R}^p \), \( u \in \mathbb{R}^q \) and \( A^*, B^*, C^* \) are constant matrices, with dimensions compatible with \( x^* \), \( u \) and \( y^* \), such that for a specified set of inputs the reduced model response is a satisfactory approximation to the original system response.

For systems described by a transfer function, the model reduction problem may be formulated as follows:

Let the high order system be given by
\[ y(s) = G(s)u(s) \quad (2.1.3) \]
where \( G(s) \) is a \( (p \times q) \) matrix transfer function of the form
\[ G(s) = \frac{A_0 + A_1 s + \ldots + A_m s^m}{b_0 + b_1 s + \ldots + b_n s^n} \quad (2.1.4) \]
where \( b_i \) \((i = 0, 1, \ldots, n)\) are scalar constants, and the \( A_i \) \((i = 0, 1, 2, \ldots, m)\) are constant \((p \times q)\) matrices. For single input-single output systems, \( p = q = 1 \), and the \( A_i \) become scalars.

The problem is to find a simplified model of a reduced order \( r < n \) given by
\[ y^* = R(s)u(s) \quad (2.1.5) \]
where \( R(s) \) is a \( (p \times q) \) matrix transfer function of the form
\[ R(s) = \frac{C_0 + C_1 s + \ldots + C_t s^t}{d_0 + d_1 s + \ldots + d_r s^r} \quad (2.1.6) \]
where \( d_i \) \((i = 0, 1, \ldots, r)\) are scalar constants and the \( C_i \)
(i = 0, 1, ..., t) are constant (pxq) matrices. The coefficients of \( R(s) \) are computed such that for a specified set of inputs, the reduced model response is a satisfactory approximation of the original system response.

In some cases, the high order system may be described by an experimental curve or by a set of input-output measurements. In these cases the model reduction problem becomes one of computing a reduced order model, of any desired order, (which may be described by a transfer function or in state-vector form) from the given input-output measurements.

2.2 Applications of Model Reduction

Model reduction techniques are of growing importance in modern control technology. They are mainly needed in the following fields:

I. The study of complex systems;

II. Optimal control systems;

III. Sensitivity analysis.

I. There has been much interest in building mathematical models of chemical processes, biological and economic systems. These give rise to systems of very high order (represented by five hundred equations, say) and this would be extremely difficult to simulate on a computer [56]. Thus, it is obviously very useful to be able to approximate this high order system by a system of lower order, which would be easy to simulate on a computer. In empirical studies of systems, the order of a system is rarely greater than fourth [28], hence a system can always be approximated very closely by a fourth (or less) order model. Kalyaev [49] used reduced order models to approximate the transient response of a high order linear time-
invariant system. In reference [75] a method of reduction was applied for the computer analysis of high order electrical circuits. In references [13, 29, 56] reduced order models were used for studies on the dynamic stability of interconnected power systems. Donati [30] used reduced order models for identifying high order systems from a set of input-output measurements. In fact the partial minimal realization problem, first introduced by Tether [93], is simply the modelling of a high order complex, possibly nonlinear system, by a reduced order constant model from a given set of input-output measurements.

II. Considerable advances have been made in the theory of optimal control during the last decade, and a number of computational methods have been developed for the solution of optimal control problems. In general, however, these methods are not suitable for on-line optimal control of complex and high order systems, due to the large amount of computation required. It is well known that the determination of the optimal control involves the solution of a non-linear matrix differential equation, the Riccati equation. In spite of the mathematical simplicity, there are certain engineering shortcomings associated with the implementation of the solution to the Riccati equation. The number of scalar equations which correspond to the matrix Riccati equation increases with the square of the order of the system equations. Thus the application of the existing design procedure to high order systems represents a computationally difficult and cumbersome task [91]. Hence there is a need for designing approximately optimal controls for high order systems, through the use of lower order models.

Many researchers have used this method of control for system design, that is designing suboptimal controllers for the system
through the use of lower order models \([91, 97]\). Nicholson \([68]\) used a fourth order reduced model of an oil-fired boiler to derive suboptimal controls for the system, while Atary \([7]\) used an 18th order reduced model to design controllers for a complete nuclear power plant prototype, whose model contains about 220 nonlinear differential equations and algebraic relations, with very encouraging results. Sinha \([91]\) derived a number of reduced models of a given system, obtained by different methods of reduction, and approximate optimal controls for them were computed. These controls were then applied to the original system and compared with the actual optimum. It was shown that the low order models provide a near-optimal control.

II. The advantages of model reduction are particularly apparent in the case of sensitivity analysis. An important problem in control engineering is the design of sensitivity functions requirements of a differential equation of at least twice the order of the system. Thus the use of low order models is of great practical significance, due to the immense savings in computer facilities. Thus a third order model of a tenth order system reduces the basic sensitivity function from twentieth order to sixth order, and can lead to sensitivity prediction being made in circumstances which might otherwise be
beyond the capacity of the computer facility. In reference [96] the transient response sensitivity of an aircraft blind landing system was studied by using a number of reduced order models. It was established that low order models adequately predict the transient response sensitivity of high order systems, leading to considerable reduction in computation time and complexity.

Towill [95] has made a short survey of model reduction methods, and has listed some advantages of using reduced order models. He has further listed some desirable features of reduction techniques, and these may vary from one problem to another.

2.3 Survey of Model Reduction Techniques

A number of different methods have been proposed during the last seven years for the reduction in order of high order dynamical systems. All the methods have the same objective: to provide a low order model of the complex system which is computationally or analytically more tractable than the complete system, yet still provides sufficient information about the system to be useful. However the approaches used are different and they may be roughly divided into the following groups:

1. Modal methods;
2. Reduction by minimisation of an error criterion;
3. Reduction by curve-fitting in the time-domain;
4. Reduction by curve-fitting in the frequency-domain;
5. Analytical methods;
6. Miscellaneous methods.

The above grouping is not meant to be a rigid one, in the sense that a method from group 2, for example, may in fact use the approaches of group 2 and group 3. The above groupings are discussed in this Section and some of the methods are discussed in more detail.
than others. This is because some of the methods are more widely used than others, and also to gain an insight of the computational requirements for various methods. In group 5, only a brief outline of the methods is given because Chapter 3 contains a thorough review of the methods (continued fraction methods and time-moments technique), and it would be superfluous to say more about them in this Chapter.

In group 6, we have included a number of methods of reduction which could not be classified in any of the other groups.

2.3.1. Modal methods

Given the exact $n$th order linear time-invariant system described by the state variable equations

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

A reduced model, of order $r$ say, is required to approximate the system of Equation (2.3.1). Nicholson [68] developed a linear model of an oil-fired boiler in state variable form, for which optimal and suboptimal controllers were designed. Nicholson found that to facilitate the computations, it was necessary to reduce the model to one of order four (the original system is of order nine). This he did by discarding the eigenvalues of the original system which are furthest away from the origin, and retained only the predominant eigenvalues of the original system in the reduced model. Davison [28] formulated the procedure for a general system as follows:

The system is transformed into a canonical form so that the eigenvalues can be distinguished. This is achieved by the normal transformation

$$x = Pz$$
where $P$ is the right hand latent matrix of $A$. The system equation then becomes
\begin{align*}
\dot{z} &= P^{-1}APz + P^{-1}Bu \\
\dot{z} &= \Lambda z + P^{-1}Bu \\
\dot{y} &= Cz
\end{align*}
(2.3.3)

where $\Lambda$ is the diagonal matrix of the eigenvalues of the system. The eigenvalues are arranged in increasing order. The transformation Equation (2.3.2) is partitioned into columns, associated with the dominant modes, and into rows, associated with the variables to be retained in the reduced model, considering the free system only at first. The forcing terms are reduced separately, by partitioning the transformed forcing matrix. Thus the $n^{th}$ order system of Equation (2.3.1) is reduced to an $r^{th}$ order system of the form
\begin{align*}
\dot{x}^* &= A^*x^* + B^*u \\
\dot{y}^* &= C^*x^*
\end{align*}
(2.3.4)

where $x^*$ is an $r$-dimensional state vector consisting of selected elements from the original state $x$. The matrices $A^*$, $B^*$ and $C^*$ are given by
\begin{align*}
A^* &= P_1^1 \Lambda_1 P_1^{-1} \\
B^* &= P_1 (P_1^{-1}B) \\
C^* &= (CP_1)P_1^{-1}
\end{align*}

where the diagonal matrix $\Lambda_1$ has the first $r$ dominant eigenvalues of $A$ as its elements. $P_1$ is the ($rxr$) matrix comprising elements of the first $r$ dominant eigenvectors of $A$, corresponding to the variables to be retained in the reduced system.

The main advantage of this method lies in its simplicity, and also the reduced order model is constrained to be stable if the given system is stable and vice versa. This is an advantage which many of the other methods do not have. However it does have a number of disadvantages such as
(a). It is not uncommon for systems to have no dominant modes;
(b). Steady state values are not necessarily the same;
(c). Phase errors may be introduced by discarding non-dominant modes;
(d). All the eigenvalues and eigenvectors of the high order system have to be computed;
(e). The method does not work for a system with a "stiff" state matrix, that is, a state matrix with a large spread of eigenvalues;
(f). It is not always possible to see which are the best states to be retained in the model [79].

Marshall [60] produced a simple modification in the method in order to ensure that the steady state values reached by the state variables under step input conditions were exactly reproduced by the model. The modification consisted of equating certain derivatives of the state variables to zero. Wilson et. al [100] extended the above method to the reduction of discrete-time systems.

Matsuburu [62] considered the problem of reducing the order of systems described by a transfer function of the form

\[ G(s) = \prod_{j=1}^{m} \frac{(1+T_j s)}{(1+T_i s)} \prod_{i=1}^{n} \frac{(1+T_i s)}{(1+T_j s)} \] (2.3.5)

Reduction is achieved by keeping the dominant poles and zeros and replacing the far away poles and zeros by an equivalent dead time. Thus the system (2.3.5) is reduced to

\[ R(s) = e^{-Ts} \prod_{j=1}^{m-p} \frac{\prod_{i=1}^{n-q} (1+T_j s)}{\prod_{i=1}^{n} (1+T_i s)} \] (2.3.6)

where

\[ T = \prod_{i=n-q-1}^{n} T_i - \prod_{j=m-p-1}^{m} T_j \]

Although this method overcomes the problem of phase errors,
Equation (2.3.6) is not very practical since the time delay has to be approximated by a Padé approximant [98] which would raise the order of the model.

The above methods suffer from the fact that they do not have a measure of "goodness of approximation". Chidambara [24] overcomes this difficulty by constraining the reduced model to be such that the integral of the squared error between the step responses of the exact and simplified models is minimised (this criterion enables a measure of goodness of approximation to be made), and also the initial and final values of the transient response of the model, under the influence of a polynomial input up to second degree in time, shall show no error when compared with the exact response. This is accomplished by making the output equation of the system (2.3.4) to be

$$\mathbf{y}^* = (C^* - \mathbf{Q}) \mathbf{x}^*$$

where \( \mathbf{Q} \) is chosen such that the reduced model will satisfy the above mentioned conditions.

Chidambara [24] further introduced a second method of reduction in which it is assumed that the eigenvalues of the reduced system are unknown and a set of nonlinear equations are derived which, when solved would give a reduced model satisfying the above mentioned constraints.

Although Chidambara has overcome some of the disadvantages of the method of Davison, it is quite clearly done at the expense of simplicity, and the method would involve excessive computation which may make the reduction procedure less desirable.

In reference [55] a generalisation of Davison's method is made so that the initial transient response of the high order system may be approximated as well as the steady state response. The method consists of dividing the total time of response into a suitable
number of time intervals; and a reduced model is evaluated for each of these time intervals. The time division is based on the relative magnitudes of the eigenvalues of the original system; and each reduced order model would contain the appropriate eigenvalues of the system. Thus the eigenvalues of the reduced model, which is to approximate the initial transient response of the system, for example, would retain the largest eigenvalues of the original system.

2.3.2. Reduction by minimisation of an error criterion

A number of methods of reduction have been developed in which the reduced order model is constrained to minimise a given cost function. The cost function is usually dependent on the error between the responses, of the high order system and the reduced model, to a given input (or class of inputs). Thus let the output of the high order system, to a given input, be \( y(t) \) and let the corresponding output of the reduced model be \( y^*(t) \). Define

\[
e(t) = y(t) - y^*(t) \tag{2.3.7}
\]

For the case of a single input-single output system, \( e(t) \) as defined in (2.3.7) will be a scalar. Then one common cost function used is of the form

\[
J = \int_0^\infty e^2(t) \, dt \tag{2.3.8}
\]

For the case of a multivariable system, with \( p \) outputs, \( e(t) \) as defined in (2.3.7) will be a \( p \)-dimensional vector, and the cost function corresponding to (2.3.8) for the multivariable case is

\[
J = \int_0^\infty e^T(t)Qe(t) \, dt \tag{2.3.9}
\]

where \( Q \) is a constant matrix which is often taken to be the identity matrix.

The coefficients of the reduced order model are then chosen to minimise the cost function \( J \). It should be noted that there are a number of different cost functions which are commonly used.
Kardashov [50] used the error criterion of Equation (2.3.8) to derive reduced order models for a given single input-single output system assuming an input of unit impulse, while Meier and Luenberger [63] assume that the input is a stationary random process. Minimisation of the cost function leads to a set of non-linear equations for the coefficients of the reduced order models. If the poles of the reduced model are prespecified, then the problem reduces to one of solving a set of linear equations. Peterka and Vidinciv [73] chose to minimise a cost function which is a function of the impulse responses of the model and the system. Fellows et. al. [34] chose to minimise the error between the step responses of the system and the reduced model. Renganathan [74] used the time-weighted square of the output and its derivatives as criterion, which penalises the error at large times.

Wilson [99] and Chidambara [24] formulated the problem in state space form with similar end results, that is the coefficients of the reduced model are given by the solution of a set of non-linear equations.

Mitra [64, 65] views the reduction process as a two stage process. In the first stage, referred to as "approximate reduction" the full model is reduced to an uncontrollable model of the same order as the original system, and whose controllable subspace is of the desired order of the reduced model. The second stage of "strict reduction" reduces the order of the uncontrollable model by selecting its controllable subspace. The first stage requires the minimisation of a projection error which is computationally very difficult [34]. The second stage consists of choosing a suitable basis of dimension $r$ (the dimension of the reduced model) and a set of $r$ variables to produce a controllable $r$th order system retaining the $r$ controllable states. The greatest disadvantage of Mitra's method
is the very large amount of computations necessary to derive the reduced order model. The paper by Fellows et. al. [34] contains a very simple exposition of Mitra's method.

Rogers and Sworder [80] redefined the problem of model reduction as follows: "For a given high order system, find a reduced order model so that the response of the system using the optimal control policy of the model is as close to that of the model as possible". Thus their method necessitates the minimisation of a certain error criterion as well as the solution of the Riccati equation.

The main advantage of the methods mentioned above is that a functional of the error between the original system and the reduced model outputs is minimised; that is, the reduced order models are chosen in accordance with a "measure of goodness" criterion. However, most of these methods have some of the following disadvantages:
(a). The methods are computationally cumbersome, since most of them require the solution of a set of nonlinear equations;
(b). The steady state responses of the system and the model are not necessarily the same;
(c). The reduced models are not necessarily stable.

2.3.3. Reduction in the time-domain

In this Section we will consider the problem of finding a reduced order model from the impulse response of a high order time-invariant system. So far there has been two approaches to this problem. The first approach involves fitting the impulse function by a model consisting of a linear combination of linearly independent functions [51, 53, 78]. The second approach consists of deriving a reduced model whose impulse response is as close to that of the system as possible [5]. A brief outline of these procedures is now given.

Suppose that the impulse response of the system is denoted by
The output \( y(t) \) and the input \( u(t) \) of the system are then related by

\[
y(t) = \int_{-\infty}^{t} g(t-\tau)u(\tau)d\tau
\]

which is the well known convolution integral. Since the transfer function of the system considered, denoted by \( G(s) \), is the Laplace transform of the impulse response, an approximation to \( G(s) \) can be obtained by fitting to \( g(t) \) a model consisting of a linear combination of functions with known Laplace transforms. Considering the case for a single input-single output system, suppose that the functions from which the reduced model of \( g(t) \), say \( r(t) \), is to be constructed are \( f_1, f_2, \ldots, f_p \); then \( r(t) \) is of the form

\[
r(t) = a_1f_1(t) + a_2f_2(t) + \ldots + a_pf_p(t)
\]

where \( a_i \) (\( i = 1, 2, \ldots, p \)) can be chosen to minimise the error function

\[
J = \int_{0}^{\infty} (g(t) - r(t))^2dt
\]

The problem of choosing \( a_i \) to minimise \( J \) is greatly simplified if the functions \( f_i \) are orthogonal, that is

\[
\int_{0}^{\infty} f_i(t)f_j(t)dt = \delta_{ij} \quad \text{(Kronecker delta)}
\]

since it is then easy to show that

\[
J = g^2(t)dt - 2\int_{0}^{\infty} \sum_{i=1}^{p} a_i g(t)f_i(t)dt + \sum_{i=1}^{p} a_i^2
\]

minimising \( J \) with respect to \( a_i \) gives

\[
a_i^* = \int_{0}^{\infty} g(t)f_i(t)dt
\]

The optimised model is then given by

\[
r(t) = \sum_{i=1}^{p} a_i^* f_i(t)
\]

and the associated approximation to the transfer function \( G(s) \) is given by

\[
R(s) = \sum_{i=1}^{p} a_i^* F_i(s)
\]
where \( F_i(s) \) is the Laplace transform of \( f_i(t) \). Chuang [27] gave a method for ensuring that the steady state values of the system and the reduced model are the same.

The advantages of this form of approximation are that the reduced order model can be assumed to be stable, and that the steady state values of the system and the model are equal. The main disadvantages of the method are that the choice of the orthogonal function is arbitrary, and that it is not easily applicable to systems described by a transfer function or a state-vector representation. It should be noted that for multivariable systems, the method is applied to individual entries of the impulse response matrix.

Anderson [5] considered the problem of deriving a reduced order model such that the difference occurring at each sampling point between the output responses of the system and the reduced model is minimised. The system Equation (2.3.1) is replaced by the set of \( n \) equivalent first order difference equations which describe the discrete-time system. These are of the form

\[
x((k+1)T) = \Phi(T)x(kT) + ATu(kT)
\] (2.3.10)

where

- \( T \) is the sampling period
- \( \Phi(T) \) is the \((nxn)\) transition matrix of the discrete system
- \( \Delta(T) \) is the \((nxq)\) driving matrix of the discrete system.

Now assuming the state vector in (2.3.10) has been arranged such that its first \( r \) elements are measurable, and denoting the vector of these elements by \( x* \), it is desired to produce a reduced system given by

\[
x*((k+1)T) = \Phi*(T)x*(kT) + \Delta*(T)u(kT)
\] (2.3.11)

where the elements of \( \Phi*(t) \) and \( \Delta*(t) \) are to be chosen such that the difference occurring at each sampling point between \( x* \) and the
first \( r \) elements of \( x(\Delta x^r) \) is minimised. In fact it would be ideal to obtain the elements of \( \mathbf{\phi}^*(T) \) and \( \Delta^*(T) \) such that \( x^r \) was a solution of Equation (2.3.11), that is

\[
x^r(T_{k+1}) = \mathbf{\phi}^*(T)x^r(kT) + \Delta^*(T)u(kT)
\]  

(2.3.12)

Anderson suggests that by expanding the \( m \)th row of (2.3.12) the following system of equations, for successive time intervals will be obtained,

\[
b_m = M c_m \quad \text{for } m = 1, 2, \ldots, r
\]  

(2.3.13)

where

\[
b_m = \{x_m^{(r)}(2T), \ldots, x_m^{(r)}((k+1)T)\}
\]

\[
c_m = \{\mathbf{\phi}^*_{m1}, \mathbf{\phi}^*_{m2}, \ldots, \mathbf{\phi}^*_{m}, \Delta^*_{m1}, \ldots, \Delta^*_{m}\}
\]

where \( \mathbf{\phi}^*_{ij} \) and \( \Delta^*_{ij} \) are the \((i,j)\) elements of \( \mathbf{\phi}^*(T) \) and \( \Delta^*(T) \) respectively. \( M \) is a matrix of dimensions \((k+1) \times (r+q)\) with an \((i+1)\)th row of the form,

\[
M_{i+1} = [x_1^{(r)}(iT), x_2^{(r)}(iT), \ldots, x_r^{(r)}(iT), u_1(iT), \ldots, u_q(iT)]
\]

However for reasonable values of \( k, r \) and \( q \) this system of equations is greatly overdetermined and there are no values of \( \mathbf{\phi}^*_{ij} \) and \( \Delta^*_{ij} \) which simultaneously satisfy all \((k+1)\) equations. But from the theory of vector spaces [35] it can be shown that if the elements of \( c_m \) are determined by the equation

\[
c_m = (M^T M)^{-1} M^T b_m,
\]

then the vector \( b^* \) which satisfies Equation (2.3.13), minimises the inner product \( \langle b_m - b^*, b_m - b^* \rangle \). If the matrices \( B \) and \( C \) are now constructed such that

\[
B = [b_1, b_2, \ldots, b_r]
\]

\[
C = [c_1, c_2, \ldots, c_{r-1}, c_r] = \{\mathbf{\phi}^*(T); \Delta^*(T)\}^T
\]

then

\[
B = MC = M\{\mathbf{\phi}^*(T); \Delta^*(T)\}^T
\]

and the least squares solution is given by

\[
C^T = \{\mathbf{\phi}^*(T); \Delta^*(T)\} = B^T M (M^T M)^{-1}
\]

Hence the elements \( \mathbf{\phi}^*(T) \) and \( \Delta^*(T) \) are computed such that the
response of the reduced system is a least squares fit of the corresponding response of the original system.

It is important to note that the time interval \((k+1)T\) over which the mean squares fit is carried out, must be chosen in such a way as to exceed the largest time constant in the system by a considerable margin. The accuracy of the reduced model depends on the properties of the matrix \(M\), which in turn depends on the interval \((k+1)T\).

Steady state accuracy increases as the time interval increases, but this time interval should not be so large as to give rise to the singularity of the matrix \(M^T M\). This usually means performing the fit over the whole transient response in order to obtain a good approximation to the original curve. However for a system which takes a comparatively long time to reach its steady state, the matrices \(M\) and \(B\) can become rather large and this would restrict the use of this method as a suitable algorithm for computer usage. In reference [79] a comparison of Anderson's method and the method of Davison is made, and the limitations of both methods are outlined by applying the methods to the reduction of some physical systems.

2.3.4. Reduction in the frequency-domain

The frequency response of a system describes its characteristics fully, and thus there are methods of reduction which attempt to make the frequency response of the reduced model approximate the frequency response of the original system as closely as possible. There are many methods of obtaining transfer functions from the frequency responses. These are

(a) Graphical methods based on the Bode plot [23, 67, 96];
(b) Dudnikov's method [31, 67];
(c) Complex curve fitting techniques [59, 71, 84, 92].
(a) The method of Kan Chen [23] relating closed loop poles to the open loop Bode plot is rapid and simple to use. Chen proposed that for model building purposes, the open loop Bode diagram could be divided into three regions

\[ |H(j\omega)|_{\text{dB}} > 15\text{dB}; \quad 15\text{dB} > |H(j\omega)|_{\text{dB}} > -15\text{dB}; \]
\[ -15\text{dB} > |H(j\omega)|_{\text{dB}} \]

and the region between +15dB and -15dB is then used to synthesise the reduced order model. The three regions correspond to the presence of dipoles near the origin, dominant poles and zeros and far-off poles and zeros respectively.

Towill [96] introduced a simple correction to Kan Chen's method which increases the accuracy of the method by taking into account the effect of the far-off poles and zeros. This is done by equating them to a single time constant.

(b) The method of Dudnikov [31] was originally a technique for fitting rather than simplifying system models. The method is thought to be one of the most powerful methods for modelling a systems transfer function from a given frequency response [14]. In this Section we will outline the method and in Chapter 3 we will establish a relationship between Dudnikov's method and the method of equating moments [58] for the reduction of linear time-invariant systems. Suppose that the transfer function, to be fitted is of the form

\[ G(s) = \frac{b_0 + b_1 s + b_2 s^2 + \ldots + b_m s^n}{1 + a_1 s + a_2 s^2 + \ldots + a_n s^n} \] (2.3.14)

Let \( s = j\omega \), and substitute in Equation (2.3.14)

\[ G(j\omega) = \frac{b_0 (j\omega)^{-n} + b_1 (j\omega)^{1-n} + \ldots + b_m (j\omega)^{m-n}}{(j\omega)^{-n} + a_1 (j\omega)^{1-n} + \ldots + a_{n-1} (j\omega)^{1-n} + a_n} \]

This is then rewritten in the form

\[ G(j\omega) = A_0 + \phi_1(j\omega) \]
in which $A_0$ is a constant, and $\phi_1$ is a function of the form

$$\phi_1(jw) = \frac{c_0(jw)^{1-n} + c_1(jw)^{2-n} + \ldots + c_{n-1}}{(jw)^{-n} + a_1(jw)^{1-n} + \ldots + a_n}$$

Now consider the function

$$G_1(jw) = \frac{1}{\phi_1(jw)}$$

which may be rewritten in the form

$$G_1(jw) = A_1 - B_1(jw)^{-1} + \phi_2(jw)$$

by separating the constant term and the term containing $(jw)^{-1}$. $\phi_2(jw)$ has a numerator of degree $(2-n)$ and denominator of degree $(1-n)$. Then continuing the procedure, outlined above, the following continued fraction for $G(jw)$ is obtained

$$G(jw) = A_0 + \frac{1}{B_0(jw)^{-1} + A_1 + \frac{1}{B_1(jw)^{-1} + \ldots + B_{n-1}(jw)^{-1} + A_n}}$$

(2.3.15)

The constants $A_0, B_0, A_1, \ldots, B_{n-1}, A_n$ are determined in the following manner [67]. Separate the real and imaginary parts of the functions as follows

$$G(jw) = P(w) + j\theta(w)$$

$$G_1(jw) = P_1(w) + j\theta_1(w)$$

$$\ldots \ldots \ldots \ldots$$

$$G_n(jw) = P_n(w) + j\theta_n(w)$$

Now as $w \to 0$

$$|G(jw)| = |P(w)| \to |A_0| = |b_0|$$

$$|G_1(jw)| = |P_1(w)| \to |B_0(jw)^{-1}|$$

$$|G_1(jw) - B_1(jw)^{-1}| \to |P_1(w)| \to |A_1|$$

$$|G_2(jw) - B_2(jw)^{-1}| \to |P_2(w)| \to |B_1(jw)^{-1}|$$

$$\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots$$

$$|G_n(jw) - B_{n-1}(jw)^{-1}| \to |P_n(w)| \to |A_n|$$

Graphs of the functions $|P_1(w)|$ and $|\theta_1(w)|$ are then used for
determining the constant coefficients $A_i$ and $B_i$.

The procedure is ended when the function $\Phi_{n+1}(j\omega)$ is zero.

Once all the coefficients are known, $G(j\omega)$ may be written in the form of Equation (2.3.15) which is then inverted to get the transfer function.

It should be noted that Equation (2.3.15) may be rewritten in the form

$$\frac{G(s) - A_0}{s} = \frac{s^{-1}}{B_0 s^{-1} + A_1 s^{-1} + \ldots + A_{n-1} s^{-1} + A_n}$$

which is known as a J-type continued fraction [98]. Reduced order models may be obtained from (2.3.15) by truncating the continued fraction and inverting.

Golant [40] described a method for using J-type continued fractions for the identification of system parameters from the high frequency part of an experimentally obtained Amplitude-Phase Characteristics curve.

(c) One of the first methods of fitting the frequency response of systems was due to Levy [59]. Suppose we have the values of $G(j\omega)$ at discrete frequencies, and let the frequency-dependent transfer function $R(j\omega)$ say, to be fitted to the generated data be

$$R(j\omega) = \frac{b_0 + b_1(j\omega) + b_2(j\omega)^2 + \ldots}{a_0 + a_1(j\omega) + a_2(j\omega)^2 + \ldots}$$

(2.3.17)

The error in the fit at frequency $\omega_k$ is

$$e_k = G(j\omega_k) - \frac{b(j\omega_k)}{a(j\omega_k)}$$

(2.3.18)

multiplying across by $a(j\omega_k)$ and separating the right hand side of the resulting equation into real and imaginary parts and squaring the absolute value gives
An error function \( J \) is now defined which is based on (2.3.19) and summed over all the frequencies considered

\[
J = \sum_{k=0}^{m} (R_k^2 + I_k^2)
\]

(2.3.20)

The method of least squares is then applied with \( J \) being differentiated with respect to each of the parameters \( b_0', b_1, b_2, \ldots, a_1, a_2, \ldots \) (\( a_0 \) being set to unity).

The method has been widely used and is capable of fitting some quite unusual frequency responses. But several difficulties have been found with the method. Sanathan [84] found that the procedure does not give a good fit if the frequency data spans several decades and has modified the method by inserting weighting factors. Thus involving the designers subjective judgement which is undesirable as it will frequently lead to errors in complex problems [22]. Numerical difficulties may also occur in Levy's method, since the least squares equations are often nearly singular.

Payne [71] modified the method by incorporating certain constraints based on additional knowledge of the system such as steady state gain, represented by the coefficient \( b_0' \). He noted that in the unconstrained system there is a tendency for the reduced model to be unstable even though the original system is stable. The addition of constraints greatly reduces this. Sumner [92] proposes a technique in which the error between the data and the model is minimised using a nonlinear optimisation method.

2.3.5. Analytic methods of reduction

Ba Hli [8] introduced a method of computing the transfer function of a system from its impulse response by fitting the moments of the
model and the system. The determination of low order models from high order systems by matching the lower moments of the impulse response of the systems and the model was suggested by Paynter [72]. The method which he used was to derive the transfer function of the system and to obtain the moments of the system and the model by a series expansion. Thus given the impulse response of a system as \( g(t) \), then

\[
G(s) = \int_0^\infty g(t)e^{-st} \, dt
\]  

(2.3.21)

then using the power series expansion for \( e^{-st} \)

\[
G(s) = \int_0^\infty g(t)\left\{1-st+s^2t^2/2! - s^3t^3/3! + \ldots\right\} \, dt
\]

\[
= \int_0^\infty g(t)dt - s\int_0^\infty tg(t)dt + s^2\int_0^\infty \frac{t^2}{2!} g(t)dt - \ldots
\]

or

\[
G(s) = c_0 + c_1s + c_2s^2 + \ldots
\]  

(2.3.22)

where

\[
c_i = \frac{(-1)^i}{i!} \int_0^\infty t^i g(t)dt, \quad \forall \ i = 0, 1, 2, \ldots
\]  

(2.3.33)

Thus the time-moments of \( g(t) \) are proportional to the coefficients of the power series expansion of \( G(s) = \mathcal{L}g(t) \), about \( s = 0 \). In the method of moments, the reduced model \( R(s) = \mathcal{L}r(l) \) is chosen so that as many as possible of its moments are equal to those of \( G(s) \). This is equivalent to choosing \( R(s) \), such that as many as possible of the coefficients of its power series expansion, about \( s = 0 \), are equal to the respective coefficients of the corresponding expansion of \( G(s) \).

In the case when \( R(s) \) is chosen to be a rational transfer function then it will be a Padé approximant of \( G(s) \) (see Appendix II).

For simplicity, throughout the thesis, the coefficients of the power series expansion of \( G(s) \) about \( s = 0 \), are referred to as the time-moments of \( G(s) \), since they are in fact proportional. It should be
noted that time-moments can only be defined for asymptotically stable systems. Hence if any of the algorithms in the thesis are being used on non-asymptotically stable systems then the term "time-moments" should be replaced by "the coefficients of the power series expansion about \( s = 0 \).

Bosley [14], and Lees [58] have used the method of moments for the modelling of chemical processes with very good results. Brown [16] used the method of moments to fit a rational transfer function, of given order \( r \), to a given sampled impulse response function.

Chen and Shieh [19] introduced a method of reduction based on the expansions of the given system transfer function into a C-type continued fraction [98], about \( s = 0 \), of the form

\[
G(s) = \frac{1}{\frac{h_1}{s} + \frac{1}{\frac{h_2}{s} + \frac{1}{\ddots + \frac{1}{\frac{s}{h_{2n}}}}}}
\]  

(2.3.24)

where \( n \) is the order of the system. A reduced model of order \( k \) is then obtained by truncating (2.3.24) after the first \( 2k \) terms. The method has been established as one of the most powerful methods for the reduction of systems [2, 14]. Chuang [26] and Horowitz [44] proposed a different method of expanding the transfer function into a continued fraction to give better approximation of the initial transient response as well as the steady state response. In Chapter 3, it will be shown that Chen's method is a special case of Padé approximation (see Appendix II). Hence Chen's method is a special case of the time-moments method (for asymptotically stable systems). This equivalence of the methods is apparently not well known in the literature.

The above methods of reduction have the following advantages:

(a). The methods are extremely simple to use;
(b). They are of general applicability (may be applied to systems described by transfer functions, state vector representation or input/output measurements);

(c). The steady state values of model and system are equal for polynomial inputs [89];

(d). They have a measure of goodness, in the sense that the time-moments of the system and model are equated up to a certain number.

It should be noted that for reduction by time-moments method, the order of the numerator of the reduced model has to be fixed prior to the reduction, while Chen and Shieh's method fixes the order of the numerator to be one less than the order of the denominator.

However the methods do suffer from a very important disadvantage; the reduced order model may be unstable even though the system is stable and vice versa.

For systems which are not asymptotically stable, the reduced models are Padé approximants, which approximate the system about $s = 0$, but it is incorrect to say that they fit the time-moments of the system, since these cannot be defined.

Gustafson [42] introduced two methods of reduction, which are only applicable to systems whose transfer functions have no numerator dynamics. The first is simply a Padé approximation method, while the second method computes the reduced model from the elements of the Routh array formed for the characteristic equation of the system. This ensures that the model has the same integral squared impulse response as the system. However this second method is restricted to systems which are asymptotically stable.

2.3.6. Miscellaneous methods of reduction

Nordahl [69] introduced a method of reduction for which the responses
of the high order system and the low order model are matched by means of a generalised energy relationship. This is accomplished by matching the hypersurfaces in the state space which are described by the Lyapunov functions of the model and the system. This assumes that the V-surfaces which have the same shape in the state space correspond to systems which are nearly identical in response. The elements of the model matrix are obtained by minimising the 'tip-angles', namely the angles between the (n-r) coordinate axes and the intersection of the hypersurface V in r-dimensions. Because of the heavy reliance on geometry, the usefulness of the method is restricted to low order systems. De Sarkar [29] overcame this problem by proposing that the coefficients of the reduced model should be computed such that the Lyapunov function \( V_r \) of the model has the property that \( \frac{V_r}{V} \) is approximately equal to \( \frac{V}{V} \), where V is the system's Lyapunov function, in r-dimensional space. However, the computations involved are still very large, which reduces the desirability of reducing the system.

Aoki [6] introduced the method of aggregation for the reduction of high order systems. It is possible to retain the predominant eigenvalues (or any other set of eigenvalues) of the high order system in the reduced model. Consider the high order system described by

\[
\dot{x} = Ax + Bu
\]

where \( x \in \mathbb{R}^n \) and \( u \in \mathbb{R}^q \). Assume that \( x \) is directly observable and that the system is controllable. Consider an r-dimensional vector \( x^* \), called the aggregated state vector, defined by

\[
x^* = Cx
\]

where \( C \) is an \((r \times n)\) constant matrix, \( r \leq n \), and assume that

\[
\text{rank } C = r
\]

Suppose

\[
\dot{x}^* = Fx^* + Gu
\]
then this implies that $F$ and $G$ in (2.3.28) are related to $A$ and $B$ by

$$
FC = CA
$$

$$
G = CB
$$

(2.3.29)

If $A$ and $C$ satisfy the matrix equation

$$
CA = CAC^T(CC^T)^{-1}C
$$

Then $F$ is given by

$$
F = CAC^T(CC^T)^{-1}
$$

(2.3.30)

The matrix $C$ is referred to as the aggregation matrix and $F$ as the aggregated matrix of $A$.

$C$ is the primary design parameter in constructing the model, and it is to be chosen in such a way that the error in reducing the system is minimised. One way of choosing $C$ is to make $F$ retain the first $r$ dominant modes (or any other set of modes) of the system matrix $A$.

Chidambara [25] uses the method of aggregation to derive a reduced order model whose output contains all the modes of the original system. This is achieved by giving the reduced model an additional input $e$, which is the difference between the output of the exact system and that of the lower order model. The state matrix of the model retains the dominant eigenvalues of the system.

This method of reduction, although it gives a good approximation, requires a very large amount of computation, and also the eigenvalues and eigenvectors of the system need to be computed.

Hsia [46] developed a method of reduction based on a simple frequency domain requirement. Let the high order system be given by

$$
G(s) = K \frac{1 + a_1s + a_2s^2 + \ldots + a_ms^m}{1 + b_1s + b_2s^2 + \ldots + b_ns^n} \quad m < n
$$

(2.3.31)

which is to be approximated by

$$
R(s) = K \frac{1 + c_1s + c_2s^2 + \ldots + c_ls^l}{1 + d_1s + d_2s^2 + \ldots + d_rs^r} \quad l < r
$$

(2.3.32)
The gain constants for both systems are the same to ensure zero steady state error of the model's step response. The coefficients $c_1$ and $d_1$ are selected in such a way that the magnitude of the frequency function $G(j\omega) / R(j\omega)$ deviates the least amount from unity for all frequencies. Thus let

$$\lambda(\omega) = \left| \frac{G(j\omega)}{R(j\omega)} \right|^2 = \lambda(0) + \lambda^2(0) \frac{\omega^2}{2!} + \lambda^4(0) \frac{\omega^4}{4!} + \ldots$$  \hspace{1cm} (2.3.33)

where $\lambda(0) = 1$ by virtue of Equations (2.3.31) and (2.3.32). Now $\lambda(\omega)$ is required to be unity hence

$$\lambda^2(0) = \lambda^4(0) = \ldots = \lambda^{2k}(0) = 0$$  \hspace{1cm} (2.3.34)

Then using the condition (2.3.34), a set of nonlinear equations are derived for the $c_1$'s and the $d_1$'s. In the case where it is required to retain the dominant modes of the original system in the reduced model, the denominator of $R(s)$ is then prespecified and only the numerator polynomial is subject to approximation. A serious disadvantage of the method is that the solution of a set of nonlinear equations is required.

2.4 Discussion and Comments

It is clear from the previous Section that a very large number of methods are available for the reduction of high order systems, and that a technique which works very well for a particular problem may break down when applied to another problem. However it is clear that the more sophisticated the technique, the greater the amount of computations needed.

The object of our work in this thesis has been to derive a method of reduction which is computationally very simple to apply, and in general produce reliable results. We have felt that if a large amount of computations have to be used in deriving the reduced order models,
then we may as well work with the original system; hence reduction becomes less desirable. Of all the methods discussed in the previous Section, it has been found that the methods of group (5) are conceptually and computationally the simplest to use. Hence in this thesis we have developed these methods and derived new results. However, the methods do have the undesirable feature of not always producing stable models for stable systems. Thus we have derived a method of reduction which retains the desirable features of the methods of groups (1) and (5), thus ensuring the simplicity and the stability of the reduced model.

It should be noted that in the study by Sinha [91], for deriving suboptimal controls using reduced order models, the method of Chen [19] is rated very highly. Similarly in the sensitivity analysis of a blind aircraft landing system [96], again the method of Chen was found to produce very good results in comparison with other methods of reduction.

Our objective has been to derive a method of reduction which satisfies the following requirements:

(I) The method is very simple to use;

(II) The method may be applied to systems described by a transfer function, in state vector form or by input-output measurements;

(III) Stable systems should result in stable models;

(IV) Steady state values should be identical for the reduced model and the system;

(V) The method must use some criteria of goodness of models.
Chapter 3
CONTINUED FRACTION METHODS FOR THE REDUCTION
OF SINGLE INPUT-SINGLE OUTPUT SYSTEMS

3.1 Introduction

A number of methods of reduction have been discussed in Chapter 2. It has been noted that most of the methods are complicated and involve a large amount of computations when compared with the time-moments method and the continued fraction method. In this Chapter, a relation will be established between the continued fraction methods, including the method of Dudnikov [31], and the time-moments method. A new continued fraction expansion is introduced which is computationally more efficient than the continued fraction expansion used by Chen and Shieh [19]. Padé approximation is used to derive equivalent reduced order models, and the advantages and disadvantages of the procedure in comparison with the continued fraction methods are discussed. Biased reduced order models are derived using continued fraction expansions. Finally, some physical justifications for using this form of approximation are given. In this Chapter, the problem of reducing the order of single input-single output systems only is considered.

In Section 3.2, the method of Chen and Shieh [19], which is based on expanding the given high order system transfer function into a C-type continued fraction, about s = 0, and truncating it after a certain number of terms, is outlined and its relation to the theory of Padé approximation stated, and hence its relation to the time-moments, assuming the system is asymptotically stable, established.

In Section 3.3, a new method of reduction is introduced which is based on the expansion of the high order system transfer function into a J-type continued fraction about s = 0. The continued fraction
may then be truncated after a certain number of terms to get the reduced model of desired order. The reduced order models obtained by this method are shown to be equivalent to those obtained by the method of Chen and Shieh, and further it is shown that the method of Dudnikov is a special case of this method. The method is computationally superior to Chen's method. J-type continued fractions are derived for systems described in state-vector form, and a general continued fraction expansion is introduced which may be used to fix the order of the numerator of the reduced model with respect to the order of the denominator.

In the method of moments and the method of Chen, the reduced model is approximating the response of the system at large times, since the expansion is made about \( s = 0 \) which corresponds to infinite time. Hence the reduced model approximates the steady state response of the system. Horowitz [44] suggested that to approximate the initial transient response of the system, the freedom of choosing the parameters should be split between large time and small time approximation. This is achieved by modifying the method of moments such that the expansions, in negative powers of \( s \), of the system and the model transfer functions will match up to a certain order. Chuang [27] introduced a method of reduction called the Modified Padé Approximation technique (MPA) which is in fact equivalent to the modification by Horowitz. Thus in Section 3.4, biased reduced order models are constructed using the Padé approximation technique. While in Section 3.5 biased reduced order models are constructed using continued fraction expansions of the system transfer function, and it is shown how an algorithm for the reduction of systems by the method of Chen, may be used twice to derive biased reduced order models. This alleviates the necessity of writing a new program.

In Section 3.6, some physical justifications for the methods
described in this Chapter are discussed. These may be classed into two groups. In the first the system is looked at from a feedback point of view. In the second it is considered in the time-domain (impulse response). Further an important characteristic, of the methods of this Chapter, relating to the commutative behaviour of the reduced order models under feedback, is outlined.

Finally in Section 3.7, two examples are considered. The first is simply to show how J-type continued fractions may be used for the reduction in order of systems. The second is chosen to illustrate the usefulness of biased reduced models.

### 3.2 Chen and Shieh’s Method of Reduction

Let the high order system transfer function be of the form

\[
G(s) = \frac{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}{A_{11} + A_{12}s + \ldots + A_{1,n+1}s^n}
\] (3.2.1)

where the \(A_{ij}\) are scalar constants. The numerator is assumed to be of order one less than that of the denominator simply to make the notation easier. The transfer function (3.2.1) is then expanded as follows

\[
G(s) = \frac{1}{A_{11} + A_{13}s + A_{32}s^2 + \ldots + A_{3n}s^n}
\]

Continuing this procedure we finally get

\[
G(s) = \frac{1}{A_{11} + \frac{A_{21}s + \ldots + A_{2n}s^n}{A_{21} + \frac{s}{A_{31} + \ldots + A_{3n}s^{n-1}}}}
\]
and this may be rewritten in the form

\[
G(s) = \frac{1}{h_1 + \frac{s}{h_2 + \frac{s}{h_3 + \ldots + \frac{s}{h_{2n}}}}}
\]  

(3.2.2)

where

\[
h_1 = \frac{A_{i,1}}{A_{i+1,1}}
\]  

(3.2.3)

and the \(\text{A}_{ij}\) \((i, j > 2)\) may be calculated using the Routh algorithm [98]. Thus using Equation (3.2.1) the following Routh array is formed

\[
\begin{array}{cccc}
A_{11} & A_{12} & A_{13} & A_{14} \\
A_{21} & A_{22} & A_{23} & A_{24} \\
A_{31} & A_{32} & A_{33} & \cdots \\
A_{41} & A_{42} & A_{43} & \cdots \\
A_{51} & A_{52} & \cdots & \cdots \\
\end{array}
\]

where

\[
A_{j,k} = A_{j-2,k+1} - A_{j-2,1} A_{j-1,k+1}/A_{j-1,1}
\]  

(3.2.4)

Once Equation (3.2.2) has been formed, simplification is affected by truncating the continued fraction after the first \(2k\) terms, to get a reduced model of order \(k\). Thus

\[
R(s) = \frac{1}{h_1 + \frac{s}{h_2 + \frac{s}{h_3 + \ldots + \frac{s}{h_{2k}}}}}
\]  

(3.2.5)

which when inverted will lead to a transfer function whose denominator and numerator are of order \(k\) and \((k-1)\) respectively. In the case when \(A_{j-1,1}\) in Equation (3.2.4) is equal to zero, then in Equation (3.2.2) \(s^2\) is factored out and the expansion is continued as before.

Wall [98] proved a theorem which connects the approximants of
(3.2.2) with the Padé table of the transfer function G(s). The theorem will be stated here, and its proof may be found in reference [98], pages 380-384.

Theorem 3.1

In a normal Padé table, the approximants of the continued fraction

\[
\frac{a_0}{1 + \frac{a_1 s}{1 + \frac{a_2 s}{1 + \cdots}}} \tag{3.2.6}
\]

where the \( a_i \) (\( i = 0, 1, 2, \ldots \)) are scalar constants, fill the stair-like sequence of squares

- \([0,0]\)
- \([0,1]\)
- \([1,1]\)
- \([1,2]\)
- \([2,2]\)
- \([2,3]\)
- \([\ldots]\)

It can be shown that the approximants of the even part (Appendix I) of (3.2.6), occupy the diagonal file of squares

- \([0,1]\)
- \([1,2]\)
- \([2,3]\)
- \([\ldots]\)

Thus it is easy to see that Equation (3.2.2) is equivalent to (3.2.6), and since Chen's method uses the even approximants of (3.2.2), then it follows that \( R(s) \) of Equation (3.2.5) is the \([k-1,k]\) Padé approximant of \( G(s) \); that is, the power series expansion of \( R(s) \), about \( s = 0 \), agrees with that of \( G(s) \) up to, and including, the term in \( s^{2k-1} \). However in Chapter 2, Section 2.3.5, it was shown that the time-moments method was equivalent to the Padé approximation technique. Hence, for asymptotically stable systems the continued fraction method of Chen and Shieh is equivalent to the method of time-moments, in the
sense that they are both special cases of Pade approximation.

3.3 Model Reduction Using J-Type Continued Fractions about s = 0

Given the transfer function (3.2.1), it may be expanded into a J-type continued fraction as follows

\[ G(s) = \frac{1}{A_{11} + \frac{A_{31}s + A_{32}s^2 + \ldots + A_{3n}s^n}{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}} \]

\[ = \frac{1}{A_{11} + \frac{A_{41}s^2 + \ldots + A_{4n}s^{n+1}}{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}} \]

where

\[ A_{3i} = \frac{A_{11}A_{2,i+1}}{A_{21}} \quad (3.3.1) \]

\[ A_{4i} = \frac{A_{31}A_{2,i+1}}{A_{21}} \quad (3.3.2) \]

continuing the expansion as above, the following continued fraction for the system is obtained

\[ G(s) = \frac{1}{A_{11} + \frac{A_{31}s + \frac{A_{41}s^2}{A_{21} + \frac{A_{51}s^2 + \ldots + A_{4n}s^{n+1}}{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}}}{A_{21} + \frac{A_{32}s^2 + \ldots + A_{3n}s^n}{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}}} \]

which may be rewritten in the form

\[ G(s) = \frac{1}{a_1 + b_1s + \frac{a_2 + b_2s + \ldots + a_n + \frac{b_n s}{s}}{s}} \quad (3.3.3) \]

where

\[ a_1 = \frac{A_{11}}{A_{21}} \]

\[ a_i = \frac{A_{2i-2,1}}{A_{2i,1}} \quad \text{for } i > 1 \quad (3.3.4) \]

\[ b_1 = \frac{A_{2i+1,1}}{A_{2i,1}} \quad \text{for } i > 1 \]
Again in the case when $A_{21,1}$ is equal to zero, then in Equation (3.3.3) $s^3$ is factored out and the expansion is continued as before.

Having expanded the system transfer function as shown in Equation (3.3.3), a reduced model, of order $k$, is obtained by truncating the continued fraction after the first $k$ terms, that is the reduced model will be given by

$$ R(s) = \frac{1}{a_1+b_1s+\cdots+s^2/a_k+b_ks} $$

which when inverted gives a transfer function whose denominator and numerator are of order $k$ and $(k-1)$ respectively. An algorithm for the inversion of continued fractions is given in Appendix I.

The reduced models obtained from (3.3.5) are equivalent to the corresponding models obtained by (3.2.5). This may be shown directly, by computing the even part of the continued fraction (3.2.5) and showing that it is the continued fraction of (3.3.5). However we will prove the equivalence of the two methods indirectly, by showing that (3.3.5) is the $[k-1,k]$ Padé approximant of $G(s)$, and hence by the uniqueness of Padé approximants, must be equivalent to the model of (3.2.5).

**Theorem 3.2**

The power series expansion, about $s = 0$, of $R(s)$ given by (3.3.5) agrees with that of $G(s)$ up to, and including the term in $s^{2k-1}$.

**Proof**

Define,

$$ A_{-1} = A_0 = 0, \quad B_{-1} = 0, \quad B_0 = 1 $$

$$ A_1 = 1 $$

$$ B_1 = a_1 + b_1s $$
and in general

\[ A_k(s) = (a_k s + b_k s) A_{k-1}(s) + s^2 A_{k-2}(s) \] (3.3.6)

\[ B_k(s) = (a_k s + b_k s) B_{k-1}(s) + s^2 B_{k-2}(s) \] (3.3.7)

The reduced model \( R(s) \), of order \( k \), is then given by

\[
R(s) = \frac{A_k(s)}{B_k(s)}
\]

We have

\[
A_{k+1} B_k - A_k B_{k+1} = (a_{k+1} s + b_{k+1} s) A_k B_k - A_k B_{k+1}
\]

\[
= A_k [(a_{k+1} s + b_{k+1} s) B_k - B_{k+1}] + s^2 [A_k B_{k-1} - A_{k-1} B_k]
\]

(3.3.8)

and using Equations (3.3.6) and (3.3.7) recursively, Equation (3.3.8)

reduces to

\[
A_{k+1} B_k - A_k B_{k+1} = (-1)^k 2^k s^k
\] (3.3.9)

Further,

\[
\frac{1}{B_{k+1}(s) B_k(s)} = \frac{1}{\beta_0^* s + \beta_1^* s^2 + \ldots}
\]

\[= \sum_{i=0}^{\infty} \alpha_i s^i
\]

(3.3.10)

Now, the difference between the two successive approximants

\[
\frac{A_{k+1}(s)}{B_{k+1}(s)} \quad \text{and} \quad \frac{A_k(s)}{B_k(s)}
\]

is given by

\[
\frac{A_{k+1}(s)}{B_{k+1}(s)} - \frac{A_k(s)}{B_k(s)} = \sum_{i=0}^{\infty} \alpha_i s^{2k+i}
\]

by using (3.3.9) and (3.3.10). Hence, since by definition

\[
G(s) = \frac{A_n(s)}{B_n(s)}
\]

we have for \( 1 \leq k \leq n \)

\[
G(s) - \frac{A_k(s)}{B_k(s)} = \sum_{i=0}^{\infty} \alpha_i s^{2k+i}
\]

\[= \sum_{i=0}^{\infty} \delta_i s^{2k+i}
\]

\[= \sum_{i=0}^{\infty} \delta_i^* s^{2k+i}
\]
Hence it follows that the power series expansions, about 
$s = 0$, of $G(s)$ and $R(s)$ agree up to the term in $s^{2k-1}$, and the 
proof is complete.

Now since the approximants of $G(s)$ given by Equation (3.3.5) 
have a numerator of order one less than the denominator [98], it 
follows by Theorem 3.2, that $R(s)$, given by Equation (3.3.5), is 
the $[k-1,k]$ Padé approximant of $G(s)$. Hence by the uniqueness of 
Padé approximants [45], $R(s)$ is equivalent to the reduced model 
given by Equation (3.2.5).

Equation (3.3.3) may be rewritten in the form

$$G(s) = \frac{s^{-1}}{a_1 s + b_1 + \frac{a_2}{s^2 + b_2 + \cdots + \frac{a_n}{s^n + b_n}}}$$

It is easy to see the similarity between the continued fraction 
used by Dudnikov (Chapter 2, Equation (2.3.16)) and that of 
Equation (3.3.11). In fact Dudnikov's method derives the $[k,k]$ 
Padé approximant of $G(s)$, and in the case when $A_0 = 0$ 
(in Equation (2.3.16), Chapter 2), then Dudnikov's method is 
equivalent to the method of this Section. It also achieves the same 
result as the time-moments method.

The $J$-type expansion does not have any advantages over the $C$-type 
expansion when the dynamics of the reduced model are considered; this 
is because both procedures produce the same reduced model. Hence the 
only advantages are computational. In fact for the reduction of systems 
described in state-vector form and for multivariable systems, the use 
of $J$-type expansions involve half the amount of computations. Also the 
inversion of $J$-type expansions is simpler than that of $C$-type 
expansions.
3.3.1 J-type expansions for systems described in state-vector form

So far we have assumed the system to be reduced is described by its transfer function. Let the system be described in the form

\[ \dot{x} = Ax + Bu \]  \hspace{1cm} (3.3.12)

\[ y = Cx \]

where \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^p \), and \( y \in \mathbb{R}^q \), and \( A, B, C \) are of compatible dimensions. The transfer function is then given by

\[ G(s) = C(sI-A)^{-1}B \]

\[ = CA^{-1}B + CA^{-2}Bs + CA^{-3}Bs^2 + \ldots \]  \hspace{1cm} (3.3.13)

Let us assume that \( A \) is non-singular, that is, the system is asymptotically stable. Rewriting (3.3.13) we get

\[ G(s) = c_0 + c_1s + c_2s^2 + \ldots \]  \hspace{1cm} (3.3.14)

where

\[ c_i = CA^{-i-1}B \]  \hspace{1cm} (3.3.15)

For single input-single output systems the \( c_i \) are constant scalars.

Equation (3.3.14) may be expanded into a J-type continued fraction as follows; let

\[ G(s) = \frac{1}{d_0 + d_1s + d_2s^2 + \ldots + d_is^i + \ldots} \]

where

\[ d_0 = c_0^{-1} \]

\[ d_1 = -c_0^{-1}c_1d_0 \]

and in general

\[ d_i = -c_0^{-1} \left( \sum_{j=0}^{i-1} c_{i-j}d_j \right) \hspace{1cm} i = 0, 1, \ldots \]  \hspace{1cm} (3.3.16)

If a reduced model, of order \( k \), is required, then the \( d_i \) are computed for \( i = 0, 1, 2, \ldots, (2k-1) \). Let

\[ G(s) = \frac{1}{d_0 + d_1s + d_2s^2G_1(s)} \]

where

\[ G_1(s) = \sum_{i=2}^{2k-1} d_is^{i-2} \]
Let
\[ G_1(s) = \frac{1}{e_0 + e_1 s + \ldots + e_i s^i + \ldots} \]
and the \( e_i \) (\( i = 0, 1, \ldots, 2k-3 \)) are computed using a relation similar to (3.3.16). Continuing this procedure we get
\[ G(s) = \frac{1}{d_0 + d_1 s + \frac{s^2}{e_0 + e_1 s + \ldots}} \]  
(3.3.17)

Hence (3.3.17) is truncated after the first \( k \) terms and inverted to give a reduced model of order \( k \). It is easy to see that if a C-type expansion is to be derived for (3.3.14), then twice the number of inversions, and hence twice the amount of computations, required for the J-type expansion have to be made.

3.3.2 Reduced order models with fixed numerator orders

One disadvantage of Chen's method is the fact that the numerator order of the reduced model is fixed to be one less than the denominator, which may lead to a bad approximation of the initial transient response, if the order of the numerator of the system transfer is much less than the order of the denominator [8]. This may be overcome by using the following type of continued fraction expansion.

Let the reduced model have a numerator whose order is an integer \( \delta \) less than its denominator. The system transfer function \( G(s) \) is then expanded as follows
\[ G(s) = \frac{1}{a_1 + b_1 s + (c_1 s^2 + \ldots + c_\delta s^\delta) + \frac{s^2}{a_2 + b_2 s + \ldots}} \]  
(3.3.18)

Reduction is then effected by truncating (3.3.18) after a certain number of terms. The reduced model will be of the form
It is easy to show that $R(s)$, given by Equation (3.3.19) is the $[k-\delta,k]$ Padé approximant of $G(s)$, by an argument similar to that of Theorem 3.2. The general continued fraction (3.3.18) may be used to generate any entry in the Padé table.

### 3.4 Padé Approximation about $s = 0$ and $s = \infty$

In Section 3.2, it has been shown that the method of reduction by continued fraction expansions is a special case of Padé approximation and that the time-moments method is also a special case of Padé approximation.

Let the transfer function of the system be of the form

$$G(s) = \frac{A_{21} + A_{22}s + \ldots + A_{2n}s^{n-1}}{A_{11} + A_{12}s + \ldots + A_{1,n+1}s^n}$$

(3.4.1)

Let the reduced model, of order $m$, be of the form

$$R(s) = \frac{a_0 + a_1s + \ldots + a_{m-1}s^{m-1} + a_ms^m}{b_0 + b_1s + \ldots + b_{m-1}s^{m-1} + b_ms^m}$$

(3.4.2)

Expand $G(s)$ into a power series about $s = 0$,

$$G(s) = c_0 + c_1s + c_2s^2 + \ldots$$

(3.4.3)

where

$$c_0 = \frac{1}{A_{11}}(A_{21})$$

$$c_1 = \frac{1}{A_{11}}(A_{22} - A_{12}c_0)$$

$$\ldots \ldots \ldots \ldots$$

$$c_{n-1} = \frac{1}{A_{11}}(A_{2n} - A_{12}c_{n-2})$$

$$c_n = \frac{1}{A_{11}}(-A_{1,n+1}c_0 + A_{12}c_1 - \ldots - A_{12}c_{n-1})$$

\[ \vdots \]

which are a recursive set of equations, and can very easily be solved for the $c_i$. For $R(s)$ to be a Padé approximant of $G(s)$, the power series expansion of (3.4.2) must agree with that of (3.4.3)
as far as, and including the term in \( s^{2m-1} \) (see Appendix II).

This leads to the following set of linear equations

\[
\begin{align*}
    a_0 &= b_0 c_0 \\
    a_1 &= b_0 c_1 + b_1 c_0 \\
    \vdots & \quad \vdots \\
    a_{m-1} &= b_0 c_{m-1} + b_1 c_{m-2} + \ldots + b_{m-1} c_0 \\
    0 &= b_0 c_m + b_1 c_{m-1} + \ldots + b_{m-1} c_1 + c_0 \\
    \vdots & \quad \vdots \\
    0 &= b_0 c_{2m-1} + b_1 c_{2m-2} + \ldots + b_{m-1} c_m + c_{m-1}
\end{align*}
\]

which serve to determine the coefficients of (3.4.2) uniquely. A solution of (3.4.4) exists if and only if the matrix defined by

\[
H = \begin{bmatrix}
    c_m & c_{m-1} & \cdots & c_1 \\
    c_{m+1} & c_m & \cdots & c_2 \\
    \vdots & \vdots & \ddots & \vdots \\
    c_{2m-1} & c_{2m-2} & \cdots & c_m
\end{bmatrix}
\]

(3.4.5)

is non-singular (N.B. \( H \) is formed from the last \( m \) equations of (3.4.4)).

The approximant \( R(s) \) as defined in Sections 3.2 and 3.3, is chosen to match \( G(s) \) at \( s = 0 \). For example, if five time-moments are matched then \( g(0) = r(0) \), \( g'(0) = r'(0) \), \ldots, \( g^{(5)}(0) = r^{(5)}(0) \) (where \( g^{(i)} \) is the \( i \)th derivative of \( g(t) \)). Thus all the available freedom is used to achieve a fit at \( s = 0 \). The point \( s = 0 \) in the \( s \)-domain is directly related to infinite time in the time-domain. Hence very good accuracy can be expected at large values of time.

Horowitz [44], suggested that to improve the response of the approximant at \( t = 0 \), the freedom of choosing the parameters of the reduced model should be split between large time and small time approximation. This may be done as follows:

Consider the Taylor series expansion of the impulse response of the high order system.
\[ g(t) = g(0) + tg'(0) + \frac{t^2}{2} g''(0) + \ldots \] (3.4.6)

the values of \( g(0), g'(0), \ldots \) etc., are very simply related to
the parameters of the transfer function \( G(s) \). Expanding \( G(s) \) in
an infinite series in powers of \( s^{-1} \), gives
\[ G(s) = \frac{d_1}{s} + \frac{d_2}{s^2} + \frac{d_3}{s^3} + \ldots \]
and taking inverse Laplace transforms of it gives
\[ g(t) = d_1 + d_2 t + d_3 t^2 + \ldots \]
\[ g(0) = d_1; \quad g'(0) = d_2; \quad g''(0) = d_3; \quad \ldots \] etc.

Thus as many degrees of freedom as may be desired can be used to
secure the desired behaviour at \( t = 0 \), leaving that number less for
approximating \( G(s) \) at \( s = 0 \). This is equivalent to the concept of
Padé approximation about two (or more) points first introduced by
Baker, et al [10], See Appendix II.

Chuang [27] formulated Horowitz's modification as follows:

Let the system transfer function (3.4.1) have the following
series expansions
\[ G(s) = c_0 + c_1 s + c_2 s^2 + \ldots \] (about \( s = 0 \)) (3.4.7)
\[ G(s) = \frac{d_1}{s} + \frac{d_2}{s^2} + \frac{d_3}{s^3} + \ldots \] (about \( s = \infty \)) (3.4.8)

Suppose \( G(s) \) is to be approximated by a reduced order model
\( R(s) \) as shown in Equation (3.4.2), such that the power series
expansion of \( R(s) \), about \( s = 0 \) (about \( s = \infty \)) is to agree with that
of Equation (3.4.7) (Equation (3.4.8)) up to the term in \( s^{m+k-1} \)
\( (s^{-m+k}) \) for some integer \( k \). Such an \( R(s) \) may be obtained by
computing its coefficients \( a_l, b_l \) using the following set of 2m
linear equations derived by equating the coefficients in (3.4.2),
(3.4.7) and (3.4.8).
\[ a_0 = b_0 c_0 \]
\[ a_1 = b_0 c_1 + b_1 c_0 \]
\[ \ldots \ldots \ldots \]
\[
\begin{align*}
    a_{m-1} &= b_0 c_{m-1} + b_1 c_{m-2} + \ldots + b_{m-1} c_0 \\
    0 &= b_0 c_m + b_1 c_{m-1} + \ldots + b_{m-1} c_1 + c_0 \\
    \vdots \quad \vdots \\
    0 &= b_0 c_{m+k-1} + b_1 c_{m+k-2} + \ldots + b_{m-1} c_k + c_{k-1} \\
    a_{m-1} &= d_1 \\
    a_{m-2} &= d_2 + b_{m-1} d_1 \\
    \vdots \quad \vdots \\
    a_k &= d_{m-k} + b_{m-1} d_{m-k-1} + \ldots + b_{k+1} d_1
\end{align*}
\]

assuming \( k \) is positive. It is obvious that the \( b_i \) \((i = 0, 1, 2, \ldots, m-1)\) in Equation (3.4.2) may be found by solving the following equation

\[
P = T b
\]  

(3.4.9)

where

\[
\begin{align*}
    b^T &= (b_0, b_1, \ldots, b_{m-1}) \\
    p^T &= (-c_{k-1}, -c_{k-2}, \ldots, -c_0, d_1, d_2, \ldots, d_{m-k})
\end{align*}
\]

\[
T = \begin{bmatrix}
    c_{m+k-1} & c_{m+k-2} & c_{m+k-3} & \ldots & c_{k+1} & c_k \\
    c_{m+k-2} & c_{m+k-3} & c_{m+k-4} & \ldots & c_k & c_{k-1} \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
    c_{m} & c_{m-1} & c_{m-2} & \ldots & c_2 & c_1 \\
    c_{m-1} & c_{m-2} & c_{m-3} & \ldots & c_1 & c_0 \\
    c_{m-2} & c_{m-3} & c_{m-4} & \ldots & c_0 & -d_1 \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
    c_{k} & c_{k-1} & c_{k-2} & \ldots & -d_{m-k-2} & -d_{m-k-1}
\end{bmatrix}
\]

Thus a unique solution exists if, and only if, the matrix \( T \) is non-singular. For \( k = m \), it is easy to see that \( T \) will reduce to (3.4.5). For \( k = 0 \), the reduced model has equal emphasis on both the steady state and initial transient response.

It should be noted that biased reduced order models may be stable, while the methods of Section 3.2 and Section 3.3 may
produce an unstable model. However this is not always the case.

3.4.1 Comparison of Padé approximation techniques with the continued fraction methods

When considering the dynamics of the reduced model, the Padé approximation technique and the continued fraction methods are equivalent. The difference occurs in the computational requirements. J-type continued fractions are in general better than the Padé approximation technique because they involve fewer computations, especially if the reduced model is of high order. Thus for a fifth order model, the Padé approximation technique would require approximately 110 operations (counting multiplications and divisions only) while the J-type fraction method requires approximately 80 operations. For a model of order ten, the difference becomes more apparent, the Padé approximation technique would require about 600 operations while the J-type fractions method requires about 300 operations.

For the reduction of systems with pure time delay, the methods of this Chapter are ideal. Since in this case, the power series expansion of the system is easily derived by multiplying the power series expansions of the transfer function and the time delay. Then Padé approximation, or J-type fraction methods, may be used to compute the reduced model transfer function.

In general for single input-single output systems, the J-type fraction method is computationally superior to the Padé approximation. Further, better numerical accuracy is obtained when using continued fractions.

3.5 Biased Reduced Models, Using Continued Fractions

Chuang [26] modified the method of Chen [19] to obtain good approximation to both the initial transient and the steady state
responses. Thus given the transfer function of Equation (3.4.1), it is expanded into a C-type continued fraction, about \( s = 0 \) and \( s = \infty \) respectively, as follows

\[
G(s) = \frac{1}{A_{11} + \frac{A_{31}s + A_{32}s^2 + \ldots + A_{3n}s^n}{A_{21} + \frac{A_{22}s + \ldots + A_{2n}s^{n-1}}{A_{12} + \frac{A_{13}s + \ldots + A_{1n}s^{n-1}}{A_{11} + \ldots}}}}
\]

where

\[
A_{31} = A_{12} - A_{11}\frac{A_{22}}{A_{21}}, \quad A_{41} = A_{21} - A_{31}\frac{A_{22}}{A_{31}}
\]

and so on...

by repeating the above sequence of expansions we have

\[
G(s) = \frac{1}{h_1 + \frac{s}{h_2 + \frac{1}{h_3 + \ldots}}}
\]

where \( h_1 = \frac{A_{11}}{A_{21}}, \quad h_2 = \frac{A_{2n}}{A_{3n}}, \quad h_3 = \frac{A_{31}}{A_{41}}, \ldots \) etc.

A reduced model of order \( m \) is then obtained by truncating (3.5.2) after \( 2m \) terms. Chuang [27] further showed that the reduced model obtained from (3.5.2) is equivalent to the model obtained from (3.4.9), with \( k = 0 \).

We propose the following procedure for obtaining biased reduced order models, based on the expansion of \( G(s) \) into a continued fraction.

Given \( G(s) \) as shown in Equation (3.4.1), a reduced model \( R(s) \),
Equation (3.4.2), which fits the first \((m+k)\) terms of (3.4.7) and the first \((m-k)\) terms of (3.4.8) may be derived by expanding \(G(s)\) into a continued fraction about \(s = 0\), as follows

\[
G(s) = \frac{1}{s + \frac{h_1}{s + \frac{h_2}{\ddots + \frac{h_{k+m}}{s + G(s)}}}}
\]  

(3.5.3)

where \(G_{k+m}(s)\) is of the form

\[
G_{k+m}(s) = \frac{\alpha_0 s + \alpha_1 s^2 + \ldots + \alpha_{n-k-m-1} s^{n-k-m}}{\beta_0 + \beta_1 s + \ldots + \beta_{n-k-m} s^{n-k-m}}
\]  

(3.5.4)

\(G_{k+m}(s)\) is then expanded about \(s = 0\), or equivalently, let \(z = s^{-1}\) and expand \(G_{k+m}(z)\) about \(z = 0\), thus

\[
G(s) = \frac{1}{s + \frac{h_1}{s + \frac{h_2}{\ddots + \frac{h_{k+m+1}}{s + \frac{h_{k+m+2}}{\ddots + \frac{h_{2n}}{s^{-1}}}}}}}
\]  

(3.5.5)

The reduced model \(R(s)\) is then given by

\[
R(s) = \frac{1}{s + \frac{h_1}{s + \frac{h_2}{\ddots + \frac{h_{k+m+1}}{s + \frac{h_{k+m+2}}{\ddots + \frac{h_{2n}}{s^{-1}}}}}}}
\]  

(3.5.6)

which when inverted gives (3.4.2).

It is easy to see from above that if an algorithm for the reduction of systems by the methods of Section 3.2 and Section 3.3 is available then by applying it twice, a biased reduced order model is obtained. Hence there is no need for writing a new algorithm for deriving biased models. For the case when \(k = 0\),
the reduced model of (3.5.6) will be the same as that derived from (3.5.2).

To show that \( R(s) \) obtained by using Equation (3.5.6) is equivalent to that derived by using (3.4.9) we have the following theorem. We will assume that \( k = 0 \) for simplicity.

**Theorem 3.3**

The power series expansion of \( R(s) \), derived from (3.5.6), about \( s = 0 \) (about \( s = \infty \)) agrees with that of Equation (3.4.7) (Equation (3.4.8)) as far as, and including the term in \( s^{m-1} \) (term in \( s^{-m} \)).

**Proof**

Consider the following general continued fraction

\[
G(s) = \frac{a_1}{b_1 + \frac{a_2}{b_2 + \frac{a_3}{b_3 + \ldots + \frac{a_{2n}}{b_{2n}}}}} \tag{3.5.7}
\]

where \( a_i, b_i \ (i = 1, 2, \ldots, 2n) \) may be functions of \( s \). Define

\[
A_{-1} = A_0 = 0, \quad B_{-1} = 0, \quad B_0 = 1
\]

\[
A_1 = a_1 \quad B_1 = b_1
\]

and in general

\[
A_p = b_p A_{p-1} + a_p A_{p-2} \tag{3.5.8}
\]

\[
B_p = b_p B_{p-1} + a_p B_{p-2} \tag{3.5.9}
\]

Then we have

\[
A_{p+2} B_p - A_p B_{p+2} = (b_p+2 A_{p+1} + a_{p+2} A_{p+1}) B_p - A_p (b_p+2 B_{p+1} + a_{p+2} B_{p+1})
\]

\[
= b_p+2 [A_p B_{p+1} - A_{p+1} B_p]
\]

\[
= -a_{p+1} b_{p+2} [A_{p+1} B_p - A_p B_{p+1}]
\]

\[
= (-1)^{p+1} b_{p+2} A_{p+2} a_2 \ldots a_p b_{p+1} \tag{3.5.10}
\]

Now we will determine the relation between the power series expansion of the approximants \( \frac{A_{2(m+1)}}{B_{2(m+1)}} \) and \( \frac{A_{2m}}{B_{2m}} \) of Equation (3.5.5), \( k = 0 \).

It is obvious that (3.5.5) may also be written in the form
\[ G(s) = \frac{1}{\frac{s^{-1}}{h_2^+} + \frac{1}{\frac{s^{-1}}{h_m^+} + \frac{1}{\frac{s^{-1}}{h_{m+1}^+} + \frac{1}{\frac{s^{-1}}{h_{m+2}^+} + \frac{1}{f}}}}}} \]

where \( f = sh_{2n} \) if \( m \) is odd
\[ = h_{2n} \text{ if } m \text{ is even.} \]

Equation (3.5.7) may be reduced to the above by putting
\[
\begin{align*}
    a_1 &= a_1^m = a_1^{m+2} = a_1^{m+3} = \ldots = 1 \\
    a_2 &= a_3 = \ldots = a_{m+1} = s \\
    b_i &= h_i, \quad i = 1, 2, \ldots, m, m+2, m+4, \ldots, q, \text{ where } q = 2n \text{ if } m \text{ is even} \\
    &= 2n-1 \text{ if } m \text{ is odd} \\
    b_i &= sh_i, \quad i = m+1, m+3, m+5, \ldots, q, \text{ where } q = 2n \text{ if } m \text{ is odd}
\end{align*}
\]

For the above \( a_i, b_i \) Equation (3.5.10) gives
\[ A_2^{m+1}B_2^{m-1} - A_2^mB_2^{m+1} = h_2^{(m+1)}s^m \]

we also have
\[ \frac{1}{f} = \frac{1}{\frac{A_2^mB_2^{m-1}A_2^{m+1}B_2^{m+1}}{B_2^{m+1}B_2^{m-1}}} = \frac{A_2^{m+1}B_2^{m-1}}{A_2^mB_2^{m+1}} \]
\[ = h_2^{(m+1)}s^m \sum_{i=0}^{\infty} a_i s^i \]
\[ = \sum_{i=0}^{\infty} (y_i) s^i \quad (3.5.11) \]

Hence it follows that the power series expansions, about \( s = 0 \), of \( A_2^{m+1}B_2^{m-1} \) and \( A_2^mB_2^{m+1} \) obtained by (3.5.5) agree up to, and including, the term in \( s^{m-1} \).

Equation (3.5.5) may be rewritten as
\[ G(s) = \frac{1}{\frac{s^{-1}}{h_2^+} + \frac{1}{\frac{s^{-1}}{h_m^+} + \frac{1}{\frac{s^{-1}}{h_{m+1}^+} + \frac{1}{\frac{s^{-1}}{h_{m+2}^+} + \frac{1}{f}}}}} \]

\[ = h_{2n} \]

and Equation (3.5.7) may be reduced to the above by putting

\[ a_i = 1, \quad i = 1, 2, \ldots, m \]
\[ a_i = s^{-1}, \quad i = m+1, m+2, \ldots, 2n \]
\[ b_1 = h_1, \quad b_2 = \frac{h_2}{s}, \quad b_3 = h_3, \quad b_4 = \frac{h_4}{s}, \ldots \]
\[ b_{m+i} = h_{m+i}, \quad i = 1, 2, \ldots, 2n-m \]

For the above \( a_i, b_i \) Equation (3.5.10) gives

\[ A_{2(m+1)}B_{2m} - A_{2m}B_{2(m+1)} = h_{2(m+1)}s^{-(m+1)} \]

we also have

\[ \frac{1}{B_{2(m+1)}B_{2m}} = \frac{1}{B_{2m}B_{2(m+1)}} \sum_{i=0}^{\infty} a_is^{-i} \]

\[ \frac{A_{2(m+1)}}{B_{2(m+1)}} \frac{A_{2m}}{B_{2m}} = \sum_{i=0}^{\infty} (\theta_i)_m s^{-(m+1+i)} \]  

(3.5.12)

Hence it follows that the power series expansions, about \( s = \infty \),

\[ \frac{A_{2(m+1)}}{B_{2(m+1)}} \quad \text{and} \quad \frac{A_{2m}}{B_{2m}} \]

obtained by (3.5.5) agree up to, and including, the term in \( s^{-m} \).

Since by definition, the original transfer function \( G(s) = \frac{A_{2n}}{B_{2n}} \),

we have for \( 1 \leq m \leq n \),

\[ G(s) - \frac{A_{2m}}{B_{2m}} = \left[ \frac{A_{2n}}{B_{2n}} - \frac{A_{2(n-1)}}{B_{2(n-1)}} \right] + \left[ \frac{A_{2(n-1)}}{B_{2(n-1)}} - \frac{A_{2(n-2)}}{B_{2(n-2)}} \right] + \ldots 
\]

\[ + \left[ \frac{A_{2(m+1)}}{B_{2(m+1)}} - \frac{A_{2m}}{B_{2m}} \right] \]

\[ = \sum_{i=0}^{\infty} (\gamma_i)_m s^{-(n+1+i)} + \ldots + \sum_{i=0}^{\infty} (\gamma_i)_m s^{m+i} \]

by using Equation (3.5.11). Similarly by using (3.5.12) we have

\[ G(s) - \frac{A_{2m}}{B_{2m}} = \sum_{i=0}^{\infty} \theta_i s^{-(m+1+i)} \]

Hence the theorem is proved.

In the analysis above, C-type continued fractions have been used so that an easy comparison may be made with existing methods [19, 26].
It is easy to show that equivalent reduced order models may be derived by using J-type continued fractions, with a saving in computations, especially if applied to the reduction of multivariable systems.

This method of computing biased reduced order models has the following advantages over the method of Padé approximation, about $s = 0$ and $s = \infty$, of Section 3.4:

I. It is a continuation of the continued fraction methods of Chen [19] and Chuang [26];

II. Slight modification of the algorithm for reduction by Chen's method is required to obtain biased models;

III. It does not fail when a particular matrix is singular (3.4.9);

IV. Computationally more efficient.

3.6 Justifications for Using Padé Type Approximations

The method of continued fractions is a well known technique in the theory of rational approximations. Truncated continued fraction approximations are often used in function evaluation routines. From the control system point of view, Chen's method of reduction may be justified as follows [19]:

Consider the general feedback system shown in Figure 3.1.

\[ \frac{Y}{u} = \frac{G}{1+GH} \]  \hspace{1cm} (3.6.1)

Dividing the numerator and denominator by $G$, we have

\[ \frac{Y}{u} = \frac{1}{H+\frac{1}{G}} \]  \hspace{1cm} (3.6.2)
Equation (3.6.2) can be considered as the simplest continued fraction expansion. The physical meaning implied in the formula is significant. It is easily seen that when $G$ is large, the overall gain can be approximated by $\frac{1}{H}$, that is, $H$ dominates the behaviour of the system.

Now consider a feedback system which has a minor feedforward loop as shown in Figure 3.2.

![Fig. 3.2 A Feedback System with a Minor Feedforward Loop](image)

The overall transfer function for Figure 3.2 is given by

\[
\frac{Y}{u} = \frac{G_{1} + F_{1}}{1 + (G_{1} + F_{1})H} = \frac{1}{H + \frac{1}{F_{1} + G_{1}}}
\]  

(3.6.3)

If the sub-system $G_{1}$ is still a high order transfer function, we can continue the expansion to get a continued fraction of the form of Equation (3.2.2), which may be rewritten in the form

\[
\frac{Y(s)}{u(s)} = \frac{1}{h_{1} + \frac{1}{h_{2} + \frac{1}{h_{3} + \frac{1}{h_{4} + \ldots + \frac{1}{h_{2n}}}}}}
\]

This corresponds to a combination of many feedback and feedforward loops as shown in Figure 3.3.
Fig. 3.3 Block Diagram Corresponding to Continued Fraction Expansion
It should be noted that the most dominant term is $h_1$ and the second influence term is $\frac{h_2}{s}$. As the quotients in the continued fraction descend to lower and lower positions they become less important as far as their influence on the performance of the system is concerned. An $m^{th}$ order model is formed by discarding all but the outer $m$ feedforward and outer $m$ feedback loops.

Another important justification for the use of the methods of this Chapter is the fact that, for asymptotically stable systems, the reduced models equate the lower moments of the system, and this may be considered as a "measure of goodness" of approximation [14]. Another interpretation of this moments-fitting procedure is that it leads to zero steady state error between the system and model outputs for polynomial inputs in time, of the form $u = \sum_{i=0}^{2m-1} a_i t^i$, where $m$ is the order of the model.

An important property of the methods of this Chapter, which relates the effect of feedback on the system and its reduced model, may be illustrated as follows:

Consider the feedback system of Figure 3.4a and 3.4b.

![Fig. 3.4a System Feedback](image)

![Fig. 3.4b Model Feedback](image)

where $G_0(s)$ is a high order system, of order $n$. $R_0(s)$ is the reduced model, of order $k$, of $G_0(s)$ (derived by the methods of this Chapter). $H(s)$ is a feedback controller of order $h$ say.

Let the closed loop transfer function of Figure 3.4a be denoted by $G_c(s)$ of order $(n+h)$, and that of Figure 3.4b be denoted by $R_c(s)$, of order $(k+h)$. Further, suppose that $R_0(s)$ fits the first $2k$
coefficients of the power series expansion of \( G_0(s) \) about \( s = 0 \) (or equivalently, for asymptotically stable systems, the first \( 2k \) time-moments of \( G_0(s) \)).

Then it can be shown that \( R_c(s) \) fits the first \( 2k \) coefficients of the power series expansion of \( G_c(s) \) about \( s = 0 \). Hence the \( k \)th order reduced model of \( G_c(s) \) is exactly the same as that of \( R_c(s) \). This is best illustrated by the following diagram

![Reduction and Feedback Process Diagram](image)

Fig. 3.5 Reduction and Feedback Process

In the special case when \( h = 0 \), that is, the feedback is a constant, then

\[ R_c'(s) = R_c(s) \]

The commutative diagram of Figure 3.5 may be shown to hold by using the feedback relation

\[ G_c(s) = \frac{G_0(s)}{1 + H(s)G_0(s)} \]

together with the power series expansions of \( G_0(s), G_c(s), R_0(s), R_c(s) \) and \( H(s) \).

The above property is very useful when the effect of feedback on the performance of a high order system is to be investigated by simulating the system and the feedback controllers on an analogue computer, in which case the order of \( G_c(s) \) may be too high for the computer. Hence a reduced model of \( G_0(s) \) may be used in the investigation.
3.7 Examples

Example 1

This example was chosen to illustrate the equivalence of reduced models derived by C-type expansions or J-type expansions. Let the high order system be given by

\[ G(s) = \frac{360 + 171s + 10s^2}{720 + 702s + 71s^2 + s^3} \]

Expanding into a C-type continued fraction gives

\[ G(s) = \frac{1}{\frac{1}{s} + \frac{5}{5 + \frac{6}{6 + \frac{4}{4}}} + 3 + \frac{5}{5 + \frac{6}{6 + \frac{4}{4}}} + 1 + \frac{s}{s}} + 2 + \frac{s}{s} \]

A reduced model of order two, is derived by truncating \( G(s) \) after the first four quotients and inverting (Appendix I)

\[ R_c(s) = \frac{15 + 6s}{30 + 27s + s^2} \]

Expanding \( G(s) \) into a J-type continued fraction gives

\[ G(s) = \frac{1}{\frac{1}{s^2} + \frac{1.2s}{1.2s + \frac{s^2}{3 + 1.2s + \frac{s^2}{150 + 11.25s^2}}} + 2 + \frac{s}{s}} + \frac{1}{s} + \frac{5}{5 + \frac{6}{6 + \frac{4}{4}}} + 3 + \frac{5}{5 + \frac{6}{6 + \frac{4}{4}}} + 1 + \frac{s}{s} \]

A reduced model of order two, is derived by truncating \( G(s) \) after the first two quotients and inverting

\[ R_j(s) = \frac{3 + 1.2s}{6 + 5.4s + 0.2s^2} \]

and multiplying the numerator and denominator by 5 gives

\[ R_j(s) = \frac{15 + 6s}{30 + 27s + s^2} = R_c(s) \]

Example 2

This example was chosen to illustrate the use of biased reduced order models for reducing high order systems. Consider the system

\[ G(s) = \frac{s^3 + 12s^2 + 54s + 72}{s^4 + 18s^3 + 97s^2 + 180s + 100} \]

Three different second order models were calculated. The model
obtained by placing all the approximation emphasis about \( s = 0 \) (that is \( k = 2 \) in Equation 3.4.9) has the transfer function

\[
R_a(s) = \frac{72+923.143s}{100+1387.143s+1346.445s^2}
\]

The model obtained by having more approximation emphasis about \( s = 0 \) than about \( s = \infty \) (that is \( k = 1 \)) was found to be unstable. The model obtained by having equal approximation emphasis about \( s = 0 \) and \( s = \infty \) (that is \( k = 0 \)) has the transfer function

\[
R_b(s) = \frac{827.273+105s}{1149+1352.273s+105s^2}
\]

The model obtained by having more approximation emphasis about \( s = \infty \) than \( s = 0 \) (that is \( k = -1 \)) has the transfer function

\[
R_c(s) = \frac{6.092+s}{8.462+11.092s+s^2}
\]

The difference between the step responses of the original system and the three models is shown in Graph 3.1.

The curve (a) represents the difference between \( G(s) \) and \( R_a(s) \), the curve (b) between \( G(s) \) and \( R_b(s) \) and the curve (c) between \( G(s) \) and \( R_c(s) \). It can be easily seen that during the initial transient period the amplitude of curve (c) is smaller than that of the curve (b) which is smaller than that of curve (a).
Graph 3.1 The Difference Between the Step Responses of the Original System and Models
Chapter 4
CONTINUED FRACTION METHODS FOR THE REDUCTION OF
CONSTANT LINEAR MULTIVARIABLE SYSTEMS

4.1 Introduction
The main purpose of this Chapter will be to extend the methods of
Chapter 3 to the problem of reducing the order of linear constant
multivariable systems. Chen [22] and Chuang [27] have introduced
methods for reducing an equal input-equal output multivariable system
using C-type continued fractions about $s = 0$, and $s = 0$ and $s = \infty$.

In this Chapter, J-type continued fraction expansions are used to
derive reduced order models of equal input-equal output multivariable
systems. The method, although giving the same models as Chen's method
(or Chuang's method) is computationally superior. Further, biased
reduced order models are derived using continued fraction expansions
about $s = 0$ and $s = \infty$. Padé approximations, about $s = 0$, and $s = 0$
and $s = \infty$ respectively are used to derive reduced order models which
are described by a set of constant differential equations. The Padé
approximations, when they exist, produce the same reduced order
models as the corresponding continued fraction expansions. Padé
approximations are, in general, computationally easier to use than
continued fractions (since the latter may require the inversion of
rational matrix transfer functions). However Padé approximations may
sometimes fail to give an answer due to the singularity of a certain
matrix, while the continued fractions may not fail in that case.
Further if an algorithm is available for reducing systems by the
method of Chen [22], then it will be shown how this algorithm may
also be used to derive biased reduced order models.

Except in Section 4.7, throughout the Chapter we will assume
that the system has an equal number of inputs and outputs. Further, we will assume that the system is described by a matrix transfer function of the form

\[ y(s) = G(s)u(s) \]  

(4.1.1)

where \( y \) and \( u \) are the output and input vectors and \( G(s) \) is the \((p \times p)\) matrix transfer function. If the system is described in state-space form as shown in Equation (4.1.2)

\[ \dot{x} = Ax + Bu \]  

(4.1.2)

\[ y = Cx + Du \]

where \( x \in \mathbb{R}^n \), \( u \) and \( y \in \mathbb{R}^p \) and the matrices \( A, B, C \) and \( D \) have dimensions compatible with \( x, u \) and \( y \), then the following relation may be used to compute the transfer function of the system

\[ G(s) = C(sI-A)^{-1}B + D \]  

(4.1.3)

Finally, the system may be described by a set of linear, constant differential equations of the form

\[ L(D)\dot{y} = M(D)u, \quad D = \frac{d}{dt} \]  

(4.1.4)

where \( L(D) \) and \( M(D) \) are \((p \times p)\) polynomial matrices. Then

\[ G(s) = [L(s)]^{-1}M(s) \]  

(4.1.5)

and hence it is assumed that \( L(s) \) is invertible [88].

In Section 4.2 a brief outline of the method of Padé approximation of analytic matrices, as expounded in reference [11], is given. In Section 4.3, an outline of the method of Chen [22] for the reduction of multivariable systems, using a matrix C-type continued fraction expansion about \( s = 0 \), is given. An alternative method of reduction is introduced which is based on the expansion of the given matrix transfer function into a J-type continued fraction expansion about \( s = 0 \). It is shown that the two methods are equivalent and that the use of J-type expansions is much more economical computationally.

In Section 4.4, the method of continued fraction synthesis of
biased reduced order models (that is, models with more emphasis on the initial transient response than the steady-state response and vice versa) as outlined in Chapter 3, for the single input-single output case, is extended to the multivariable case.

In Section 4.5 it is shown how reduced order models may be derived by using the concept of Padé approximation about \( s = 0 \), and about \( s = 0 \) and \( s = \infty \). A comparison is made between the use of continued fraction methods and Padé approximation methods for the reduction of multivariable systems.

In Section 4.6 an example is given to show how the above methods may be applied.

Finally, in Section 4.7 the methods of Padé approximation are extended to the problem of reducing a general multivariable system.

### 4.2 Padé Approximation of Analytic Matrices

Given a formal power series expansion

\[
G(s) = \sum_{i=0}^{\infty} C_i s^i
\]

where the set of \( \{C_i\} \) is a set of constant square matrices. Since the matrices form a non-commutative algebra, we can define two types of Padé approximants [11]. If the denominator is placed on the right-hand-side (left-hand side respectively), we call the approximant the right-hand side approximant (left-hand approximant respectively) thus

\[
G(s) = P_N(s)[Q_M(s)]^{-1} + O(s^{M+N+1})
\]

\[
= [Q_M(s)]^{-1}P_N(s) + O(s^{M+N+1})
\]

where \( P_N(s), \hat{P}_N(s) \) and \( Q_M(s), \hat{Q}_M(s) \) are polynomials, in \( s \), of degree \( N \) and \( M \) respectively, and with matrices for coefficients. It can be shown that the left-hand side and right-hand side approximants are equivalent as follows:

We replace

\[
A(s) = B(s) + O(s^{M+N+1})
\]
by

\[ A(s) \equiv B(s) \quad (4.2.5) \]

we have

\[ P_N(s)[Q_N(s)]^{-1} = [\hat{Q}_M(s)]^{-1} P_N(s) + O(s^{M+N+1}) \quad (4.2.6) \]

hence

\[ \hat{Q}_M(s)P_N(s) - \hat{P}_N(s)Q_M(s) = O(s^{M+N+1}) \quad (4.2.7) \]

However a polynomial of degree \((N+M)\) which is of order \(s^{M+N+1}\) is identically zero. Therefore

\[ P_N(s)[Q_N(s)]^{-1} = [\hat{Q}_M(s)]^{-1} P_N(s) \quad (4.2.8) \]

Therefore if a function has a right-hand and left-hand side \([N,M]\) Padé approximant, then they are equal to each other.

Left-hand and right-hand side continued fraction expansions of analytic matrices can be defined as follows

\[ f_n(s) = I + sA_n [f_{n+1}(s)]^{-1} \quad (4.2.9) \]

\[ g_n(s) = I + s[g_{n+1}(s)]^{-1} B_n \quad (4.2.10) \]

where \(f_n(s)\) and \(g_n(s)\) are analytic matrices and \(A_n, B_n\) are constant square matrices. These so defined expansions are called the expansion in continued fraction near the origin of a function \(G(s)\), and

\[ g_1(s) = f_1(s) = G(s), \quad [11]. \]

Thus given a matrix analytic function \(G(s)\), we can expand it into a right-hand side continued fraction of the form

\[ G(s) = \frac{B_0}{I + \frac{sB_1}{I + \frac{sB_2}{\ddots + \frac{sB_n}{}}}} \quad (4.2.11) \]

where \(I\) is the identity matrix.

From Equations (4.2.9) and (4.2.10), it can be shown that a continued fraction cut at a certain order is equal to an \([N,N]\) or \([N,N+1]\) Padé approximant. More generally, a continued fraction expansion may be expressed in the form
\[ f_n(s) = A_n + s [f_{n+1}(s)]^{-1} \]

from which it is easy to construct the right-hand and left-hand side approximants of type \([N,N]\) or \([N,N+1]\). The denominators and the numerators of the approximants satisfy the recurrence Equations (4.2.13) - (4.2.16). Thus given the following continued fraction expansion of a matrix transfer function

\[
G(s) = \frac{I}{A_1 + \frac{\text{si}}{A_2 + \frac{\text{si}}{A_3 + \ldots}}} \tag{4.2.12}
\]

The numerators and denominators of the right-hand side Padé approximants are given by

\[
P_{N+1} = P_N A_{N+1} + s P_{N-1} \tag{4.2.13}
\]
\[
Q_{N+1} = Q_N A_{N+1} + s Q_{N-1} \tag{4.2.14}
\]

and for the left-hand side approximants we have

\[
P_{N+1} = A_{N+1} P_N + s P_{N-1} \tag{4.2.15}
\]
\[
Q_{N+1} = A_{N+1} Q_N + s Q_{N-1} \tag{4.2.16}
\]

In this Chapter we will be mainly concerned with left-hand side approximants and thus will often refer to Equations (4.2.15) and (4.2.16).

4.3 Model Reduction Using Continued Fraction Expansions about \(s = 0\)

Suppose that the high order system is given by

\[ y(s) = G(s)u(s) \tag{4.3.1} \]

where \(u\) and \(y\) are the (px1) input and output vectors respectively, and \(G(s)\) is the system matrix transfer function.

Chen's method of reduction consists of expanding \(G(s)\) into a matrix C-type continued fraction, about \(s = 0\), of the form

\[
G(s) = [H_1^1 + sH_2^1 + sH_3^1 + \ldots]^{-1} \tag{4.3.2}
\]
where $H_1, H_2, \ldots$, are constant (pxp) matrices.

It is obvious that Equation (4.3.2) may be rewritten in the form

$$G(s)^{-1} = H_1 + s[H_2 + s[H_3 + s[\ldots]^{-1}]^{-1}]^{-1}$$  \hspace{1cm} (4.3.3)

provided $[G(s)]^{-1}$ exists [88]. $H_1$ can then be determined by equating the constant matrices on both sides of the above equation. Similarly $H_2, H_3, \ldots$, may be determined by continuously inverting and equating the constant matrices. Reduction is effected by truncating (4.3.2) after a certain number of terms and inverting to obtain a reduced order model. Thus a reduced order model which retains the first four terms of (4.3.2) is given by

$$R_4(s) = [H_1 + s[H_2 + s[H_3 + s[H_4]^{-1}]^{-1}]^{-1}]^{-1}$$  \hspace{1cm} (4.3.4)

Equation (4.3.4) may be rewritten in matrix continued fraction form as follows

$$R_4(s) = \frac{I}{H_1 + \frac{SI}{H_2 + \frac{SI}{H_3 + \frac{SI}{H_4}}}}$$  \hspace{1cm} (4.3.5)

which can be seen to be of the same form as (4.2.12) and hence using (4.2.15) and (4.2.16) we can write $R_4(s)$ in the form

$$R_4(s) = [\hat{Q}_4(s)]^{-1}\hat{P}_4(s)$$  \hspace{1cm} (4.3.6)

Then using the algorithm of Chapter 5, $\hat{Q}_4(s)$ is inverted and multiplied by $\hat{P}_4(s)$ to give a rational matrix transfer function for $R_4(s)$. The above method of computing $R_4(s)$ is more economical than continually inverting (4.3.4). It should be noted that the approximants of (4.3.2) are the $[k-1,k]$ and the $[k,k]$ Padé approximants of $G(s)$.

We propose the following equivalent method of reduction. Given the system of Equation (4.3.1), we expand $G(s)$, about $s = 0$, into a matrix $J$-type continued fraction of the form
\[ G(s) = [A_1 + B_1 s + \frac{s^2}{[A_2 + B_2 s + \frac{s^2}{[\ldots]}]} \ldots]^{-1} \]  \hspace{1cm} (4.3.7)

where \( A_i, B_i \) \((i = 1, 2, 3, \ldots)\) are constant \((p \times p)\) matrices.

It is obvious that (4.3.7) may be rewritten in the form
\[
[G(s)]^{-1} = A_1 + B_1 s + \frac{s^2}{A_2 + B_2 s + \frac{s^2}{[\ldots]}}^{-1}
\]  \hspace{1cm} (4.3.8)

provided that \([G(s)]^{-1}\) exists. \( A_1 \) and \( B_1 \) can then be determined by equating coefficients on both sides of Equation (4.3.8). Then let
\[
G_1(s) = \left( [G(s)]^{-1} - (A_1 + B_1 s) \right) / s^2
\]  \hspace{1cm} (4.3.9)

and hence
\[
[G_1(s)]^{-1} = A_2 + B_2 s + \frac{s^2}{A_3 + B_3 s + \frac{s^2}{[\ldots]}}^{-1}
\]  \hspace{1cm} (4.3.10)

Thus provided \([G_1(s)]^{-1}\) exists, then \( A_2 \) and \( B_2 \) can be determined by equating coefficients from both sides of Equation (4.3.10); \( A_3, B_3, A_4, B_4, \ldots \), may be computed similarly. It will be shown in Section 4.3.1 that the approximants of Equation (4.3.7) form the \([k-1,k]\) Padé approximant of \( G(s) \). Thus a reduced order model which is equivalent to (4.3.4) is given by
\[
R_4(s) = [A_1 + B_1 s + \frac{s^2}{[A_2 + B_2 s]}]^{-1}
\]  \hspace{1cm} (4.3.11)

Equation (4.3.11) may be rewritten in matrix continued fraction form as
\[
R_4(s) = \frac{I}{(A_1 + B_1) + \frac{S^2 I}{A_2 + B_2 s}}
\]

which can be seen to be of the same form as (4.2.12) and hence using (4.2.15) and (4.2.16) we can write \( R_4(s) \) in the form
\[
R_4(s) = [L_4(s)]^{-1} M_4(s)
\]  \hspace{1cm} (4.3.12)

where \( L_4(s) \) and \( M_4(s) \) are polynomial matrices.

It is very easy to see that deriving the \([k-1,k]\) Padé approximant by using C-type continued fractions involves \(2k\) inversions of rational transfer function matrices, while using J-type continued fractions involves only \(k\) inversions of rational transfer function matrices. Hence the use of J-type expansions is computationally much more attractive.
4.3.1 Equivalence of the two methods

From Section 4.2 it can be seen that the approximants derived from (4.3.2) are the \([k-1,k]\) Padé approximants of \(G(s)\), that is, the power series expansion of the reduced model transfer function \(R(s)\), about \(s = 0\), agrees with that of \(G(s)\) up to and including the term \(s^{2k-1}\) for some \(k\). This has in fact been shown in reference [27].

In this section we will show that the approximants derived from (4.3.4) by truncating the continued fraction after \(k\) terms is the \([k-1,k]\) Padé approximant of \(G(s)\). Then by the uniqueness of the Padé approximants, the equivalence of the two methods is established, if it is assumed that (4.3.2) is truncated after \(2k\) terms.

Theorem 4.1

The power series expansion, about \(s = 0\), of the reduced order model obtained by truncating (4.3.7) after \(k\) terms agrees with the corresponding power series expansion of \(G(s)\) up to, and including, the term in \(s^{2k-1}\); that is, the reduced model is the \([k-1,k]\) Padé approximant of \(G(s)\).

Proof

The rational system transfer function matrix \(G(s)\) is assumed to have a \(J\)-type-continued fraction expansion of the form

\[
G(s) = \left[ A_1 + B_1 s + s^2 A_2 + B_2 s + s^2 [... -1] -1 \right]^{-1}
\]

and its power series expansion is given by

\[
G(s) = C_0 + C_1 s + C_2 s^2 + ... \quad \text{(expanded about } s = 0)\]

where \(C_i \ (i = 0, 1, ..., )\) are constant \((p \times p)\) matrices. Define

\[
P_{-1} = P_0 = 0, \quad Q_{-1} = 0, \quad Q_0 = I
\]

\[
P_1 = I
\]

\[
Q_1 = A_1 + B_1 s
\]

\[
P_2 = A_2 + B_2 s
\]

\[
Q_2 = (A_2 + B_2 s)(A_1 + B_1 s) + s^2 I
\]
and in general (see Equations (4.2.15) and (4.2.16))

\[ P_k(s) = (A_k + B_k s)P_{k-1}(s) + s^2 P_{k-2}(s) \]
\[ Q_k(s) = (A_k + B_k s)Q_{k-1}(s) + s^2 Q_{k-2}(s) \]

The reduced model \( R(s) \), of order \( k \), is then given by

\[ R_k(s) = [Q_k(s)]^{-1}P_k(s) \]

A low order model \( R_{k+1}(s) \) can be written as

\[
R_{k+1}(s) = [Q_{k+1}(s)]^{-1}P_{k+1}(s)
\]
\[
= \{(A_{k+1} + B_{k+1} s)Q_k(s) + s^2 Q_{k-1}(s)\}^{-1}\{(A_{k+1} + B_{k+1} s)P_k(s) + s^2 P_{k-1}(s)\}
\]
\[
= \{(A_{k+1} + B_{k+1} s)Q_k(s) + s^2 Q_{k-1}(s)\}^{-1}\{(A_{k+1} + B_{k+1} s)Q_k(s) + s^2 Q_{k-1}(s)\}^{-1}\{(A_{k+1} + B_{k+1} s)Q_k(s) + s^2 Q_{k-1}(s)\}
\]
\[
= \{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}^{-1}\{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}^{-1}\{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}
\]
\[
= \{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}^{-1}\{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}^{-1}\{A_{k+1}Q_k^{-1}(s) + s^2 Q_{k+1}(s)\}
\]

Thus the difference between two successive models \( R_{k+1}(s) \) and \( R_k(s) \) is given by

\[
R_{k+1}(s) - R_k(s) = -s^2 Q_{k+1}(s)Q_{k-1}(s)[R_k(s) - R_{k-1}(s)]
\]

By repeating the same procedure on \([R_k(s) - R_{k-1}(s)]\) and so on, we get

\[
R_{k+1}(s) - R_k(s) = (-1)^k Q_{k+1}(s)Q_{k-1}(s)Q_k(s)\ldots Q_2(s)Q_0(s)R_1(s)s^{2k}.
\]

Since \( Q_i(s) \) (\( i = 0,1,\ldots \)) are polynomial matrices, it is easy to show that

\[
\{Q_{k+1}(s)Q_{k-1}(s)Q_k(s)\ldots Q_2(s)Q_0(s)\} R_1(s) = \sum_{i=0}^{\infty} L_i s^i
\]

where the \( L_i \) are constant matrices. Hence

\[
R_{k+1}(s) - R_k(s) = \sum_{i=0}^{\infty} (-1)^k (L_k) s^{2k+i}
\]

Now by definition, the original matrix transfer function is given by

\[
G(s) = Q_n^{-1}(s)P_n(s) \quad \text{(for some } n)\]

Hence the difference between \( G(s) \) and \( R_k(s) \) is given by

\[
G(s) - R_k(s) = R_n(s) - R_k(s)
\]
\[ R_n(s) = (R_n(s) - R_{n-1}(s)) + (R_{n-1}(s) - R_{n-2}(s)) + \ldots + (R_{k+1}(s) - R_k(s)) \]
\[ = \sum_{i=0}^{\infty} \frac{(L_{n-1})_i s^{2n-1+i}}{i!} + \sum_{i=0}^{\infty} \frac{(L_{n-2})_i s^{2n-3+i}}{i!} + \ldots + \sum_{i=0}^{\infty} \frac{(L_k)_i s^{2k+i}}{i!} \]
\[ = \sum_{i=0}^{\infty} \frac{V_i s^{2k+i}}{i!} \]
\[ G(s) = R_k(s) + O(s^{2k}) \]

Thus we can conclude that the power series expansion about \( s = 0 \), of the reduced order model \( R_k(s) \) obtained by truncating (4.3.7) agrees with the corresponding power series expansion of \( G(s) \) up to, and including, the term in \( s^{2k-1} \).

### 4.3.2 Continued fraction expansions for systems described by their time-moments

Consider the system whose transfer function matrix is given by
\[ G(s) = C_0 + C_1 s + C_2 s^2 + \ldots \] (4.3.13)
where the \( C_i \) are \((p \times p)\) constant matrices. The system of Equation (4.3.13) may be expanded into a \( J \)-type continued fraction using the same procedure as in Section 3.3.1, Chapter 3, but with a difference. The difference being that the \( C \)'s in this section are matrices. Thus let
\[ G(s) = [D_0 + D_1 s + D_2 s^2 + \ldots]^{-1} \] (4.3.14)

Then
\[ D_0 = C_0^{-1} \]
\[ D_1 = -D_0 C_1 C_0^{-1} \]
and in general
\[ D_i = -(\sum_{j=0}^{i-1} D_j C_{i-j})C_0^{-1} \] (4.3.15)

The order of multiplication in (4.3.15) is important, since the \( D \) and \( C \) matrices are not necessarily commutative. It is clear that \( C_0 \) has to be non-singular for this procedure to work. If \( C_0 \)
is singular, then the algorithm of Chapter 5 may be used on the
infinite series (4.3.13) to invert \( G(s) \). Let
\[
G_1(s) = \left[ [G(s)]^{-1} - \left( D_0 + D_1 s \right) \right] / s^2
= D_2 + D_3 s + D_4 s^2 + \ldots
\]  
(4.3.16)
and further let
\[
G_1(s) = \left[ D_2 + D_3 s + D_4 s^2 + \ldots \right]^{-1} = [E_0 + E_1 s + E_2 s^2 + \ldots]^{-1}
\]
where the \( E_i \) matrices are computed using a relation similar to
(4.3.15). Again this depends on the non-singularity of \( D_2 \). If
\( D_2 \) is singular then the algorithm of Chapter 5 is applied to the
power series (4.3.16) and \( G_1(s) \) is inverted. Thus now let
\[
G_2(s) = \left[ [G_1(s)]^{-1} - (E_0 + E_1 s) \right] / s^2
= E_2 + E_3 s + E_4 s^2 + \ldots
\]  
(4.3.17)
and continuing the above procedure we get
\[
G(s) = \frac{I}{D_0 + D_1 s + \frac{D_2 + D_3 s + D_4 s^2 + \ldots}{E_0 + E_1 s + \ldots}}
\]  
(4.3.18)
The continued fraction (4.3.18) is then truncated and inverted
to get the desired order model.

The above procedure for generating continued fractions is the
one to be followed whatever the description of the original system
may be. Since, if in the above algorithm \( C_0, D_2, E_2, \ldots \) were all
invertible, then the computational requirements for generating a
reduced order model would be negligible. However, if a number of
the matrices \( C_0, D_2, E_2, \ldots \) were singular then the use of the
inversion algorithm of Chapter 5 would increase the amount of
computation required.

It should be noted that we have assumed that \( G_1(s), G_2(s) \ldots \)
are invertible.

It is easy to see that the use of C-type expansions would
involve at least twice the amount of computation, and in certain
cases, the C-type expansions may require the inversion of a rational matrix while the J-type fraction would not.

4.4 Model Reduction Using Biased Continued Fraction Expansions

Given the high order system transfer function matrix $G(s)$, as shown in Equation (4.3.1), Chuang [27] expands $G(s)$ into a matrix C-type continued fraction about $s = 0$ and $s = \infty$ alternately, as follows

$$G(s) = \left[H_1 + s[H_2 + s[H_3 + s[H_4 + \ldots]]^{-1}]^{-1}\right]^{-1}$$

(4.4.1)

where $H_1, H_2, \ldots$, are constant (p×p) matrices. Reduction is then effected by truncating (4.4.1) after a certain number of terms and inverting. Thus a reduced order model which retains the first four terms of (4.4.1) is given by

$$R_4(s) = \left[H_1 + s[H_2 + s[H_3 + s[H_4 + \ldots]]^{-1}]^{-1}\right]^{-1}$$

(4.4.2)

which may be rewritten in matrix continued fraction form as follows

$$R_4(s) = \frac{I}{H_1 + \frac{sI}{H_2 + \frac{sI}{H_3 + \frac{sI}{H_4}}}}$$

(4.4.3)

which can be inverted to get $R_4(s)$ in the form

$$R_4(s) = L_4^{-1}(s)M_4(s)$$

In particular Chuang has shown that the power series expansion, about $s = 0$ ($s=\infty$), of a reduced model $R_{2k}(s)$ obtained from (4.4.1), agrees with the corresponding power series expansion of $G(s)$ up to, and including, the term in $s^{-1}(s^{-k})$.

Computationally Chuang's approach would necessarily require the inversion of $2k$ rational matrices, while the approach to be outlined may only require the inversion of one rational matrix.

We propose the following general continued fraction method for
deriving biased reduced order models. Given $G(s)$, as shown in Equation (4.3.1), it is expanded into a continued fraction expansion about $s = 0$, for the first $k$ terms and then about $s = \infty$ for the next $\ell$ terms. Thus

$$G(s) = \left[ H_1 + s[H_2 + s[H_3 + \ldots + s[H_k + G_k(s)]^{-1}]]^{-1} \right]^{-1}$$

(4.4.4)

$$= \left[ H_1 + s[H_2 + s[H_3 + \ldots + s[H_k + s[H_{k+1} + s[H_{k+2} + \ldots \right. \]

$$\left. + s[H_{k+\ell} + \ldots ]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right)$$

(4.4.5)

A reduced order model $R_{k+\ell}(s)$ whose power series expansion about $s = 0$ and $s = \infty$ agree with the corresponding power series expansion of $G(s)$ up to, and including, the term in $s^{-k-1}$ and $s^{\ell}$ respectively, is given by

$$R_{k+\ell}(s) = \left[ H_1 + s[H_2 + s[H_3 + \ldots + s[H_k + s[H_{k+1} + s[H_{k+2} + \ldots \right. \]

$$\left. + s[H_{k+\ell} + \ldots ]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right]^{-1} \right)$$

(4.4.6)

which can then be inverted to give

$$R_{k+\ell}(s) = L_{k+\ell}(s)M_{k+\ell}(s)$$

It should be noted that $(k+\ell)$ must be even to ensure that the polynomial matrices $L_{k+\ell}(s)$ and $M_{k+\ell}(s)$ are of degree $\frac{k+\ell}{2}$ and $\frac{k+\ell}{2} - 1$ respectively.

To show that the power series expansion about $s = 0$ ($s=\infty$) of $R_{k+\ell}(s)$ agrees with that of $G(s)$ up to the term in $s^{-k-1}$ ($s^{\ell}$), a proof similar to that given in Section 4.3 is carried out.

For the case when $k = \ell$, the above method gives the same reduced order models as those obtained by Chuang's method. This then implies that the order of the expansion about $s = 0$ and $s = \infty$ is unimportant. Hence it is very easy to see how an algorithm developed for the reduction of systems by Chen's method or the J-type method may be applied twice to compute biased reduced order models.

We have so far given the C-type continued fraction method for computing biased reduced order models. The corresponding J-type expansion is obtained by expanding $G(s)$ about $s = 0$ for $\frac{k}{2}$ terms and
then about \( s = \infty \) for the next \( \frac{k}{2} \) terms. Thus

\[
G(s) = [A_1 + B_1 s + s^2 [A_2 + B_2 s + \cdots + s^2 [A_k + B_k s + G_k(s)]^{-1} \cdots ]^{-1}]^{-1}
\]

(4.4.7)

\[
= [A_1 + B_1 s + s^2 [\cdots + s^2 [A_k + B_k s + G_k(s)]^{-1} \cdots ]^{-1}]^{-1}
\]

(4.4.8)

A reduced order model \( R_{k+\ell}(s) \) whose power series expansions about \( s = 0 \) and \( s = \infty \) agree with that of \( G(s) \) up to, and including, the term in \( s^{k-1} \) and \( s^{-\ell} \) respectively is given by

\[
R_{k+\ell}(s) = [A_1 + B_1 s + s^2 [\cdots + s^2 [A_k + B_k s + G_k(s)]^{-1} \cdots ]^{-1}]^{-1}
\]

(4.4.9)

which can then be inverted to give

\[
R_{k+\ell}(s) = L_{k+\ell}^{-1}(s) M_{k+\ell}(s)
\]

4.5 Padé Approximation of Multivariable Systems

So far it has been shown how a multivariable system with an equal number of inputs and outputs may be reduced by using continued fraction expansions. It has also been shown that these methods are special cases of Padé approximation. In this Section reduced order models are derived using Padé approximation about \( s = 0 \), and about \( s = 0 \) and \( s = \infty \).

4.5.1 Padé Approximation about \( s = 0 \)

Suppose that the system is given in any of the forms (4.1.1), (4.1.2) or (4.1.4). Then the system transfer function may be written in the form

\[
G(s) = \sum_{i=0}^{\infty} C_i s^i
\]

(4.5.1)
where the $C_i$ ($i = 0, 1, \ldots$) are constant (pxp) matrices.

Let the reduced order model be described by a set of constant differential equations

$$L_R(s)y(s) = M_R(s)u(s)$$

or

$$R_R(s) = L_R^{-1}(s)M_R(s)$$

where

$$L_R(s) = L_0 + L_1s + \ldots + L_{r-1}s^{r-1} + Is^r$$

and

$$M_R(s) = M_0 + M_1s + \ldots + M_\ell s^\ell$$

where $L_i$ ($i = 0, 1, \ldots, r-1$) and $M_i$ ($i = 0, 1, \ldots, \ell$) are constant (pxp) matrices.

If $R_R(s)$ is to approximate $G(s)$ in the Padé sense, then its power series expansion, about $s = 0$, must agree with that of (4.5.1) as far as and including the term in $s^{r+\ell-1}$. Hence equating coefficients we get

$$M_0 = L_0C_0$$

$$M_1 = L_0C_1 + L_1C_0$$

$$\ldots \ldots \ldots \ldots$$

$$M_\ell = L_0C_\ell + L_1C_{\ell-1} + \ldots + L_\ell C_0$$

$$O = L_0C_{\ell+1} + L_1C_\ell + \ldots + L_{\ell+1}C_0$$

$$\ldots \ldots \ldots \ldots \ldots \ldots$$

$$O = L_0C_r + L_1C_{r-1} + \ldots + L_{r-1}C_1 + C_0$$

$$\ldots \ldots \ldots \ldots \ldots \ldots$$

$$O = L_0C_{r+\ell} + L_1C_{\ell+r-1} + \ldots + C_\ell$$

and rewriting in matrix form gives
Equation (4.5.7) can then be used to solve for the $L_i$ ($i = 0, 1, \ldots, r-1$) and the $M_i$ ($i = 0, 1, \ldots, k$) uniquely.

The last $q$ equations of (4.5.7) may be rewritten in the form

\[
\begin{bmatrix}
C_{r+1} & C_r & C_{r-1} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
C_{r-1} & C_r & C_{r-2} & \cdots & C_0 \\
C_r & C_{r-1} & C_{r-2} & \cdots & C_1 \\
C_{r+2} & C_{r+1} & C_{r+2} & \cdots & C_{l+1}
\end{bmatrix}
\begin{bmatrix}
L_0 \\
L_1 \\
\vdots \\
L_{r-1}
\end{bmatrix} =
\begin{bmatrix}
C_l \\
- C_0 \\
\vdots \\
- C_l
\end{bmatrix}
\tag{4.5.8}
\]

or

\[
TL = C \\
L = T^{-1}C
\tag{4.5.9}
\]

Hence for a Padé approximant to exist, the matrix $T$ must not be singular.

In the special case when

\[
l = r - 1
\tag{4.5.10}
\]

the Padé approximant would produce the same reduced order models as
those obtained by Chen [22] or by using J-type continued fractions as shown in Section 4.3.

4.5.2 Padé approximation about \( s = 0 \) and \( s = \infty \)

Let the system have the following power series expansion

\[
G(s) = \sum_{i=0}^{\infty} C_i s^i \quad \text{(expansion about } s = 0) \tag{4.5.11}
\]

and

\[
G(s) = \sum_{i=0}^{\infty} D_i s^{-i} \quad \text{(expansion about } s = \infty) \tag{4.5.12}
\]

Let the reduced order model be described by a set of differential equations of the form

\[
L_R(s)y(s) = M_R(s)u(s) \tag{4.5.13}
\]

or

\[
R_R(s) = L_R^{-1}(s)M_R(s) \tag{4.5.14}
\]

where

\[
L_R(s) = L_0 + L_1 s + \ldots + L_{r-1} s^{r-1} + is^r \tag{4.5.15}
\]

and

\[
M_R(s) = M_0 + M_1 s + \ldots + M_{r-1} s^{r-1} \tag{4.5.16}
\]

where \( L_i (i = 0, 1, \ldots, r-1) \) and \( M_i (i = 0, 1, \ldots, r-1) \) are constant (p×p) matrices. For simplicity of notation we have assumed that \( M_R(s) \) is of order \( (r-1) \).

Suppose that \( R_R(s) \) is to approximate \( G(s) \) such that the power series expansion of \( R_R(s) \) about \( s = 0 \) \((s = \infty)\) will agree with the corresponding series of \( G(s) \) as far as the term in \( s^{k-1}(s^{-\ell}) \), for some \( k \) and \( \ell \). Then equating coefficients (noting that \( k + \ell = 2r \)) and assuming \( D_0 = 0 \),

\[
M_{r-1} = D_1
\]

\[
M_{r-2} = L_{r-1}D_1 + D_2
\]

\[
\ldots
\]

\[
M_{r-\ell} = L_{r-\ell-1}D_1 + L_{r-\ell-2}D_2 + \ldots + D_\ell
\]
\( M_0 = L_0 C_0 \)  
\( M_1 = L_0 C_1 + L_1 C_0 \)  
\( \vdots \) \( \vdots \) \( \vdots \)  
\( M_k = L_0 C_k + L_1 C_{k-1} + \ldots + L_k C_0 \)  
assuming that \( k > \ell \) and  
\( M_j = 0 \ \forall \ j > r - 1 \)  

Equation (4.5.17) consists of \( 2r \) simultaneous matrix equations which can be uniquely solved to determine the coefficients of (4.5.15) and (4.5.16). The matrix transfer function of the reduced model can then be determined by inverting \( L_R(s) \), using the algorithm of Chapter 5 and substituting in (4.5.14). By rewriting Equation (4.5.17) it is easy to show that a solution does not exist if the following matrix \( H \) is singular.

\[
H = \begin{bmatrix}
D_{\ell-1} & D_{\ell-2} & \cdots & D_1 & -C_0 & -C_1 & \cdots & -C_{r-\ell-1} \\
D_{\ell-2} & D_{\ell-3} & \cdots & -C_0 & -C_1 & -C_2 & \cdots & -C_{r-\ell} \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
-C_0 & -C_1 & \cdots & \cdots & \cdots & \cdots & -C_{r-1} \\
-C_1 & -C_2 & \cdots & \cdots & \cdots & \cdots & -C_r \\
\vdots & \vdots & \cdots & \vdots & \cdots & \cdots & \cdots & \vdots \\
-C_{r-\ell-1} & -C_{r-\ell} & \cdots & \cdots & \cdots & \cdots & -C_{2r-\ell}
\end{bmatrix}
\]

(4.5.19)

In the case when \( k = \ell \), the reduced order model is the same as that obtained by Chuang [27].

4.5.3 **Comparison of Padé approximation and continued fraction methods**

As far as the dynamics of the reduced order models are concerned, the Padé approximation techniques and the continued fraction methods are equivalent. Computationally, however, the methods are different. It should first be stated that the C-type expansions are inferior to the J-type expansions and should therefore not be used for reducing
multivariable systems. Hence a comparison will be made between J-type expansions and Padé approximation techniques.

The Padé approximation, in general, is computationally much easier than continued fraction methods, but the Padé approximation may fail to give a reduced model of desired order if the matrix $T$, in Equation (4.5.9) or equivalently for biased models the matrix $H$ of Equation (4.5.19), is singular. In that case J-type expansions may exist and thus a reduced model of desired order obtained.

For unbiased models, if the J-type expansion can be obtained by using the method outlined in Section 4.3.2 without having to invert rational matrices, then it would be computationally equivalent to the Padé approximation technique.

For biased reduced order models, if the matrix $H$ is non-singular, then the Padé approximation is computationally easier than the continued fraction methods, since the latter would necessarily involve the inversion of a rational matrix.

Thus in general for the reduction of equal input-equal output multivariable systems, the Padé approximation technique, if it produces a model, is computationally simpler than the J-type continued fraction methods.

It should be noted that for unique reduced models, the order of the model is proportional to $p$.

### 4.6 Example 1

Consider the following high order system matrix transfer function

\[
G(s) = \begin{bmatrix}
\frac{2(s+5)}{(s+1)(s+10)} & \frac{(s+4)}{(s+2)(s+5)} \\
\frac{(s+10)}{(s+1)(s+20)} & \frac{(s+6)}{(s+2)(s+3)}
\end{bmatrix}
\]
Padé approximation is used to derive two reduced order models. The first model 'fits' the first four terms of the power series expansion of $G(s)$ about $s = 0$ (that is, it equates the four time-moments of the system and the model). The second model 'fits' the first two terms of the power series expansions of $G(s)$ about $s = 0$ and $s = \infty$.

Approximation about $s = 0$

Expand $G(s)$ about $s = 0$ to give
\[
G(s) = C_0 + C_1 s + C_2 s^2 + C_3 s^3 + \ldots \tag{4.6.1}
\]
where
\[
C_0 = \begin{bmatrix} 1.0 & 0.4 \\ 1.0 & 0.5 \end{bmatrix}, \quad C_1 = \begin{bmatrix} -0.9 & -0.18 \\ -0.475 & -0.6667 \end{bmatrix},
\]
\[
C_2 = \begin{bmatrix} 0.89 & 0.086 \\ 0.47375 & 0.38889 \end{bmatrix}, \quad C_3 = \begin{bmatrix} -0.889 & -0.0422 \\ -0.47369 & -0.21296 \end{bmatrix},
\]
etc. Let
\[
R(s) = (B_0 + B_1 s + Is^2)^{-1} (A_0 + A_1 s)
\]
where $A_0$, $A_1$, $B_0$, and $B_1$ are constant $(2\times2)$ matrices. Since $R(s)$ is to approximate $G(s)$ in the Padé sense (see Section 4.5.1), we have
\[
A_0 = B_0 C_0
\]
\[
A_1 = B_0 C_1 + B_1 C_0
\]
\[
B_0 = B_0 C_2 + B_1 C_1 + C_0
\]
\[
B_1 = B_0 C_3 + B_1 C_2 + C_1
\]
The last two matrix equations can then be solved to give
\[
B_0 = \begin{bmatrix} 6.8451 & 5.4891 \\ -0.7075 & 7.1969 \end{bmatrix}
\]
\[
B_1 = \begin{bmatrix} 9.6661 & 2.0740 \\ 0.9173 & 5.6484 \end{bmatrix}
\]
Then using the first two matrix equations leads to
\[
A_0 = \begin{bmatrix} 9.5896 & 8.2271 \\ 2.8927 & 6.9146 \end{bmatrix}
\]
\[
A_1 = \begin{bmatrix}
1.9352 & 1.0489 \\
0.9581 & 1.3445
\end{bmatrix}
\]

But

\[
R(s) = (B_0 + B_1 s + Is^2)^{-1} (A_0 + A_1 s)
\]

\[
R(s) = \left(\frac{1.935s^3 + 18.533s^2 + 56.835s + 53.137}{s^4 + 15.318s^3 + 67.723s^2 + 104.658s + 53.137} \right)
\]

\[
(1.049s^3 + 11.363s^2 + 32.298s + 21.255)(1.3445s^3 + 18.948s^2 + 69.233s + 53.137)
\]

\[
(0.958s^3 + 10.379s^2 + 27.089s + 26.568)
\]

The poles of \(R(s)\) are given by

\[
s_{1,2} = -1.225 \pm 0.1714j
\]

\[
s_3 = -3.855
\]

\[
s_4 = -9.01
\]

The step responses of \(R(s)\) and \(G(s)\) were plotted as shown in Graphs 4.1 and 4.2.

**Approximation about \(s = 0\) and \(s = \infty\)**

Expand \(G(s)\) about \(s = \infty\) to give

\[
G(s) = \frac{D_1}{s} + \frac{D_2}{s^2} + \ldots
\]

where

\[
D_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}
\]

\[
D_2 = \begin{bmatrix} -12 & -3 \\ -11 & 1 \end{bmatrix}
\]

Let \(R_b(s) = (\hat{B}_0 + \hat{B}_1 s + Is^2)^{-1} (\hat{A}_0 + \hat{A}_1 s)\)

where \(\hat{A}_0, \hat{A}_1, \hat{B}_0,\) and \(\hat{B}_1\) are constant (2x2) matrices. \(R_b\) is required to approximate \(G(s)\) such that its power series expansion about \(s = 0\) (about \(s = \infty\)) will agree with that of \(G(s)\) up to, and including, the term in \(s^{-1}(s^{-2})\). Thus equating coefficients we get the following set of matrix equations

\[
\hat{A}_0 = \hat{B}_0 C_0.
\]

\[
\hat{A}_1 = \hat{B}_0 C_1 + \hat{B}_1 C_0
\]

\[
\hat{A}_1 = D_1
\]

\[
\hat{A}_0 = D_2 + \hat{B}_1 D_1
\]
which can be reduced to
\[ \hat{B}_0 C_0 - \hat{B}_1 D_1 = D_2 \]
\[ \hat{B}_0 C_1 + \hat{B}_1 C_0 = D_1 \]
which can be solved for \( \hat{B}_0 \) and \( \hat{B}_1 \) to give
\[
\hat{B}_0 = \begin{bmatrix}
10.47803 & 7.48164 \\
2.67204 & 15.24971
\end{bmatrix}
\hat{B}_1 = \begin{bmatrix}
13.7960 & 2.37685 \\
5.97837 & 9.34015
\end{bmatrix}
\]
Hence
\[
\hat{A}_0 = \begin{bmatrix}
14.21885 & 11.67285 \\
10.29690 & 16.31853
\end{bmatrix}
\hat{A}_1 = D_1 = \begin{bmatrix}
2 & 1 \\
1 & 1
\end{bmatrix}
\]
\[
R_b(s) = \frac{(2s^3+30.522s^2+131.359s+139.796)(s^3+18.636s^2+78.008s+55.912)}{(s^3+12.136s^2+62.184s+69.898)(s^3+24.136s^2+163.152s+139.796)}
\]
\[
s_1 = -1.01 \\
s_2 = -1.61 \\
s_3 = -5.75
\]
and
\[
s_4 = -14.75
\]
The responses of \( R(s) \) and \( G(s) \) were plotted and were found to be very close together. Hence the plot has not been reproduced.

4.7 Padé Approximations for the Reduction of a General Multivariable System

The continued fraction methods are restricted to the reduction of equal input-equal output multivariable systems. In this section it will be shown how Padé approximation may be used for the reduction of a general multivariable system. Consider the system given by
\[
y(s) = G(s)u(s) \quad (4.7.1)
\]
where \( y \in \mathbb{R}^p \) and \( u \in \mathbb{R}^q \) and \( G(s) \) is a \((p \times q)\) matrix transfer function.

Let
\[
G(s) = C_0 + C_1 s + C_2 s^2 + \ldots \quad (4.7.2)
\]
where \( C_i \) \((i = 0, 1, 2, \ldots)\) are constant \((p \times q)\) matrices. It is well known [27] that the Padé approximation technique when applied to (4.7.2) does not, in general, produce a unique reduced order model unless the correct order of \( R(s) \) is chosen. Let the reduced model \( R(s) \) be described by a set of constant differential equations of the form
\[
L_R(s)y(s) = M_R(s)u(s) \quad (4.7.3)
\]
where
\[
L_R(s) = L_0 + L_1 s + \ldots + L_{k-1} s^{k-1} + Is^k \quad (4.7.4)
\]
and
\[
M_R(s) = M_0 + M_1 s + \ldots + M_{k-1} s^{k-1} \quad (4.7.5)
\]
where \( L_i \) \((i = 0, 1, \ldots, k-1)\) are constant \((p \times p)\) matrices and the \( M_i \) \((i = 0, 1, \ldots, k-1)\) are constant \((p \times q)\) matrices, and \( I \) is the \((p \times p)\) identity matrix.

Let the number of time-moments of (4.7.2), to be fitted, be \( x \). Then for a unique reduced order model, the order \( k \) has to satisfy the relation
\[
k \ (p \times p) = (p \times q) \ (x-k) \quad (4.7.6)
\]
Thus the number of time-moments have to be chosen such that \( k \) as given by Equation (4.7.6) is an integer. Thus the least \( k \) is chosen for the largest possible \( x \). This is best illustrated by an example.

**Example 2**

Consider the system described by the transfer function
\[ G(s) = \frac{(s+20)}{(s+1)(s+10)} \]

Expanding \( G(s) \) into a power series gives

\[ G(s) = \left(2 - \frac{2.1}{s} - \frac{2.11}{s^2} - \frac{2.111}{s^3} + \ldots\right) \]

In this case \( p = 2, q = 1 \)

therefore

\[ k = \frac{qx}{p+q} = \frac{x}{3} \]

hence

\[ x = 3 \Rightarrow k = 1. \]

Thus let \( R(s) \) be given by

\[ (L_0 + Is)Y(s) = M_0 Y(s) \]

where \( L_0 \) is a \((2x2)\) constant matrix and \( M_0 \) is a \((2x1)\) constant matrix.

Then for \( R(s) \) to be a Padé approximant of \( G(s) \) we have

\[ M_0 = L_0 C_0 \]
\[ 0 = L_0 C_1 + C_0 \]
\[ 0 = L_0 C_2 + C_1 \]

Then using the last two equations we have

\[ L_0 C_1 = -C_0 \]
\[ L_0 C_2 = -C_1 \]

therefore

\[ L_0 \begin{bmatrix} C_1 & C_2 \end{bmatrix} = -\begin{bmatrix} C_0 & C_1 \end{bmatrix} \]

or

\[ L_0 P = -Q \]

where \( P = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \) and \( Q = \begin{bmatrix} C_0 & C_1 \end{bmatrix} \), and \( P \) and \( Q \) are constant \((2x2)\) matrices. Thus solving for \( L_0 \) gives

\[ L_0 = -Q P^{-1} \]

\[ = \begin{bmatrix} 2 & -2.1 \\ 1 & -0.6 \end{bmatrix} \begin{bmatrix} -2.1 & 2.11 \\ -0.6 & 0.32 \end{bmatrix}^{-1} \]
\[ M = L C \]

\[ R(s) = [sI + L_0]^{-1} M_0 \]

\[ R(s) = [sI + L_0]^{-1} M_0 \]

\[ R(s) = \frac{A_0 + A_1 s + \ldots + A_{k-1} s^{k-1}}{B_0 + B_1 s + \ldots + B_{k-1} s^{k-1}} \]

Rissanen [76] introduced another generalization of the Padé approximation to the case of multivariable systems. The method differs from the preceding one in that \( L_R(s) \) is taken to be a scalar polynomial. That is, the reduced model is of the form

\[ R(s) = \frac{A_0 + A_1 s + \ldots + A_{k-1} s^{k-1}}{B_0 + B_1 s + \ldots + B_{k-1} s^{k-1}} = \frac{M_R(s)}{L_R(s)} \]  

where the \( A_i \) (\( i = 0, 1, \ldots, k-1 \)) are constant \((p \times q)\) matrices, and the \( b_i \) (\( i = 0, 1, \ldots, k-1 \)) are constant scalars. It was concluded, however, that for a unique approximant to exist, the degree of \( L_R(s) \) in Equation (4.7.6) is proportional to the product \((p \times q)\) and grows rapidly when \( p \) and \( q \) are large, and thus this form of approximation is less attractive than that of the type used in this chapter, where the order of the characteristic equation of the model is proportional to the greater of the numbers \( p, q \).
System response

Reduced model response

Graph 4.1  Output 1
System response

Reduced model response

Graph 4.2 Output 2
Graph 4.3

Output 1

System response

Reduced model response

Graph 4.4

Output 2

System response

Reduced model response
CHAPTER 5
CONSTRUCTION OF THE INVERSE OF LINEAR TIME-ININVARIANT MULTIVARIABLE SYSTEMS

5.1 Introduction

Questions regarding the invertibility of dynamical systems arise in many investigations. Anderson and Moore [4] have determined a "whitening filter" which is the dynamical inverse of a dynamical model for a random process; Behn and Ho [12] have considered a simple form of inversion in the determination of pursuer or evader strategies in a stochastic game; Falb and Wolovich [33] have discussed the implications of invertibility for linear feedback decoupling; Godbole and Smith [39] use an inverse for designing a control scheme; and in Chapter 4 the use of continued fraction methods for the reduction of multivariable systems may require the continuous inversion of rational transfer function matrices, and the use of Padé approximation methods lead to reduced models which are described by a set of differential equations, and thus it is necessary to invert polynomial matrices.

Despite its apparent interest, the invertibility of dynamical systems has only recently received attention in the literature. The general problem of constructing inverses was considered by Youla and Dorato [103]. Silverman [88] has studied the existence and construction of inverse systems by methods similar to those of Youla and Dorato but in greater detail. Sain and Massey [83] introduced an algorithm for constructing the inverse by establishing an essential relationship between the system and an associated sequential circuit. Singh and Lin [90] have made a survey of the various methods of inversion.

In this Chapter a new algorithm for computing the inverse of a system is introduced. The algorithm may be applied to systems...
described in state-space form or by a matrix transfer function or by a set of constant differential equations. The algorithm is based on the theory of Padé approximation.

Thus in Section 5.2, Padé approximation is used to determine the transfer function of a system from its power series expansions. In Section 5.3 a method is introduced for computing the transfer function of a system described in state-space form. Section 5.4 contains the definition of an inverse system, and a criterion for systems to be invertible.

In Section 5.5 an algorithm for inverting systems with an equal number of inputs and outputs is introduced. The algorithm is conceptually very simple and computationally very easy. It incorporates a very simple criterion which is part of the inversion algorithm, for determining whether a system is invertible. The method may be applied to continuous or discrete-time systems. In Section 5.6 the algorithm is extended to multivariable systems with more outputs than inputs. Several illustrative examples are discussed throughout the chapter.

5.2 Construction of the System Transfer Function from its Power Series Expansions.

In Section 4.3.2, continued fraction expansions were derived by using the power series expansion of the high order system transfer function. It was shown that if any of a number of matrices was singular, then an algorithm for the inversion of systems has to be used. The first step however is to reconstruct a transfer function from the power series expansion. Also in the inversion algorithm of Sections 5.5 and 5.6, it is necessary to derive the transfer function of a system from its power series expansion. In this Section the problem of reconstructing the system transfer function from its
power series expansion is considered. Two cases are treated. The first is when the order of the numerator is known to be less than or equal to the order of the denominator. This case is useful when inverting a system by using its Markov parameters. The second case is when the order of the numerator is unknown with respect to the order of the denominator. This is useful when inverting a system by using its time-moments.

Theorem 5.1

If the function \( f(x) \) given by

\[
f(x) = C_0 + C_1 x + C_2 x^2 + \ldots
\]

reduces to a rational function \( \frac{P_m(x)}{Q_n(x)} \) given by

\[
\frac{P_m(x)}{Q_n(x)} = \frac{a_0 + a_1 x + \ldots + a_m x^m}{b_0 + b_1 x + \ldots + b_{n-1} x^{n-1} + x^n}
\]

then the following recursive relationship must hold,

\[
C_k = -\sum_{j=1}^{m} b_j C_{k-j}, \quad \forall k > m
\]

Proof

It should be noted that in the statement of the theorem \( b_n \) is normalised to unity and that

\[
C_k = 0, \quad \forall k < 0
\]

The proof of the theorem is very simple and relation (5.2.3) can be shown to hold simply by considering the power series expansion of a given rational function. Thus let

\[
\frac{P_m(x)}{Q_n(x)} = C'_0 + C'_1 x + C'_2 x^2 + \ldots
\]

Then using Equation (5.2.2), we get

\[
C'_0 = \frac{a_0}{b_0}
\]

and in general

\[
C'_k = \frac{1}{b_0} \left( a_k - \sum_{j=1}^{m} b_j C'_{k-j} \right), \quad \forall k > 0
\]
with \( a_k = 0 \ \forall k > m \). Hence Equation (5.2.5) reduces to

\[
C'_k = -\frac{1}{b_0} \sum_{j=1}^{k} b_j C'_{k-j}, \quad \forall k > m
\]

and since

\[
f(x) = \frac{P_m(x)}{Q_n(x)}
\]

then it follows that \( C'_k = C_k \), and hence the Theorem is proved.

Now consider the problem when the power series expansion of \( f(x) \)
is given by Equation (5.2.1) and we would like to obtain two polynomials.

\[
P_m(x) = \sum_{i=0}^{m} a_i x^i
\]

and

\[
Q_n(x) = \sum_{i=0}^{n} b_i x^i
\]

such that

\[
f(x) = \frac{P_m(x)}{Q_n(x)}
\]

Define the following Hankel matrices

\[
H_{U,P} = \begin{bmatrix}
C_{U-p+1} & C_{U-p+2} & \cdots & C_U \\
C_{U-p+2} & C_{U-p+3} & \cdots & C_{U+1} \\
& & \ddots & \cdots \\
C_U & C_{U+1} & \cdots & C_{U+p-1}
\end{bmatrix}
\]

then we have the following theorem [45].

**Theorem 5.2**

If the power series of Equation (5.2.1) represents the quotient of two polynomials of exact degree \( m \) and \( n \), then the matrix \( H_{m+U,n+U} \) is singular for every \( U \geq 1 \) and \( U' \geq 1 \) but is non-singular for \( U = 0 \) or \( U' = 0 \) or both.

Consider the following two cases.

**m \leq n**

Using Theorems 5.1 and 5.2, it follows that the order of the system, that is \( n \), is given by
Thus having determined the order of the system, the coefficients of \( Q_n(x) \) are given by [see Appendix II]

\[
H^*_{n,n} \mathbf{b} = -\mathbf{c}
\]

(5.2.11)

where

\[
\mathbf{b}^T = [b_{n-1}, b_{n-2}, \ldots, b_0]
\]

\[
\mathbf{c}^T = [c_1, c_2, \ldots, c_n]
\]

and

\[
H^*_{n,n} = \begin{bmatrix}
C_2 & C_3 & \cdots & C_{n+1} \\
C_3 & C_4 & \cdots & C_{n+2} \\
\vdots & \vdots & \ddots & \vdots \\
C_{n+1} & C_{n+2} & \cdots & C_{2n}
\end{bmatrix}
\]

therefore

\[
\mathbf{b} = -(H^*_{n,n})^{-1}\mathbf{c}
\]

The coefficients of \( P_m(x) \) are given by

\[
\begin{bmatrix}
a_0 \\
a_1 \\
\vdots \\
a_m
\end{bmatrix} = \begin{bmatrix}
a_0 \\
0 \\
\vdots \\
0
\end{bmatrix} \times \begin{bmatrix}
b_0 \\
b_1 \\
\vdots \\
b_m
\end{bmatrix}
\]

(5.2.12)

and

\[
a_{m+i} = 0 \quad \forall \ i > 0
\]

\( m > n \)

In this case, we are assuming that \( m \) is greater than \( n \), in fact this case should be used whenever the relative order of \( m \) to \( n \) is unknown. Let
The rank of \( L_{i,i}(C_i) \) is computed for \( i = 1, 2, \ldots, n_1 \), where

\[
\begin{align*}
L_{i,i}(C_i) &= \begin{bmatrix} C_1 & C_2 & \cdots & C_i \\ C_2 & C_3 & \cdots & C_{i+1} \\ \vdots & \vdots & \ddots & \vdots \\ C_i & C_{i+1} & \cdots & C_{2i-1} \end{bmatrix} \\
\end{align*}
\] (5.2.13)

The rank of \( L_{i,i}(C_i) \) is computed for \( i = 1, 2, \ldots, n_1 \), where

\[
n_1 = \text{rank } L_{n_1,n_1}(C_1) = \text{rank } L_{n_1+1,n_1+1}(C_1) \quad \forall \mathbf{u}, \mathbf{y} \geq 0
\] (5.2.14)

if \( m \leq n \), then \( n = n_1 \).

It is easy to show, using Theorem 5.1 that if \( m > n \), then

\[
\text{rank } L_{n_1,n_1}(C_2) < n_1
\] (5.2.15)

Thus if \( n_1 \) does satisfy the inequality of (5.2.15), then the rank of \( L_{n_1-1,n_1-1}(C_2) \) is computed. Let

\[
n_2 = \text{rank } L_{n_1-1,n_1-1}(C_2) = \text{rank } L_{n_2,n_2}(C_2)
\]

The procedure is continued until equality holds in Equation (5.2.15). Then \( n = n_1 \) (where \( i = m-n \)), and Theorems 5.1 and 5.2 may be used to compute the coefficients of \( P_m(x) \) and \( Q_n(x) \).

The power series expansion (5.2.1) may be the power series expansion of the system transfer function, about \( s = 0 \), that is, \( s = x \), in which case the \( C_i \) (\( i = 0, 1, 2, \ldots \)) are proportional to the time-moments of the system, assuming the system is asymptotically stable. If (5.2.1) is the power series expansion of the system transfer function about \( s = \infty \), that is, \( s = x^{-1} \), then the \( C_i \) (\( i = 0, 1, 2, \ldots \)) are the Markov parameters of the system.

5.3 To Compute the Transfer Function of a System Described in State-Space Form

Consider the following linear time-invariant dynamical system.

\[
\dot{x}(t) = Ax(t) + Bu(t)
\] (5.3.1)
\[ y(t) = Cx(t) + Du(t) \]

where \( u \) is the \( q \)-dimensional input vector, \( y \) is the \( p \)-dimensional output vector and \( x \) is the \( n \)-dimensional state vector. The matrices \( A, B, C \) and \( D \) have dimensions compatible with \( x, u \) and \( y \).

We are very often interested in the frequency response of the system and thus have to obtain its transfer function. By taking Laplace transforms of Equation (5.3.1) it can be shown that

\[ y(s) = [C(sI-A)^{-1}B + D]u(s) \]

\[ = G(s)u(s) \quad (5.3.2) \]

where

\[ G(s) = [C(sI-A)^{-1}B + D] \quad (5.3.3) \]

\( G(s) \) is a rational transfer function matrix of dimension \((p \times q)\).

Various methods are available for computing \( G(s) \) [35]. These involve the computation of the characteristic matrix \((sI-A)^{-1}\) using Fadeeva's algorithm and then using Equation (5.3.3) to compute \( G(s) \).

Lal and Singh [57] have proposed a method of computing \( G(s) \) by first determining \( \text{det}[sI-A] \), which gives the common denominator of each entry of \( G(s) \), while the numerator of each entry is given in terms of the Markov parameters of the system. In this Section a method is proposed for computing \( G(s) \) directly from the Markov parameters of the system. The method has the advantage of giving each entry of \( G(s) \) in reduced form and also it is directly applicable to systems described by input/output data.

Equation (5.3.3) may be rewritten in the form

\[ G(s) = C(sI-A)^{-1}B + D \]

\[ = D + C[s^{-1}I+s^{-2}A+s^{-3}A^2+\ldots]B \]

\[ = D + CBS^{-1} + CAB^{-2}s^{-2} + CA^2Bs^{-3} + \ldots \quad (5.3.4) \]

\[ = D_0 + D_1s^{-1} + D_2s^{-2} + \ldots \quad (5.3.5) \]

where

\[ D_0 = D \]
The $D_i$'s are the so-called Markov parameters of the system.

In the case when $p = q = 1$, $D_i$ ($i = 0, 1, \ldots$) are constant scalars and $G(s)$ is simply an infinite series in negative powers of $s$. Since the system (5.3.1) is a finite dimensional system, then this power series for $G(s)$ is equivalent to a rational transfer function. Then it is very easy to see that by letting $z = s^{-1}$, Equation (5.3.5) is then similar to Equation (5.2.1) and thus the algorithm of Section 2 is used to construct the unique rational transfer function for $G(s)$.

For a multivariable system, $G(s)$ will be a rational transfer function matrix and the $D_i$ ($i = 0, 1, 2, \ldots$) will be constant ($p \times q$) matrices. In this case the analysis above for the single input-single output case is carried out for each entry of $G(s)$.

This method has the following advantages:

(i) Each entry of $G(s)$ will be in reduced form;

(ii) The method may be applied to systems described by their input/output parameters (Markov parameters and time-moments) without having to derive a state-space realization;

(iii) Computationally very simple to apply, especially if $p$ and $q$ are small.

We have so far considered systems which are described in a state-space form. However systems are not always in state-space form, but may be described by a set of constant differential equations of the form

$$L(D)y = M(D)u, \quad D = \frac{d}{dt}$$

(5.3.7)

where $L(D)$ and $M(D)$ are ($p \times p$) and ($p \times q$) polynomial matrices respectively.

The transfer function of the system (5.3.7) is given by

$$y(s) = L^{-1}(s)M(s)u(s)$$
where
\[ G(s) = L^{-1}(s)M(s) \] (5.3.9)

The transfer function \( G(s) \) can be obtained by inverting \( L(s) \), using the algorithm to be outlined, and postmultiplying by \( M(s) \). More directly the Markov parameters (or the time-moments) of system (5.3.7) are obtained as outlined in Section 6.3, and then the method of Section 5.5.2 may be used to compute the transfer function.

Example 1
Consider the following system [57]

\[
A = \begin{bmatrix} 0 & -2 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}
\]

\[
C = \begin{bmatrix} 2 & 0 & 0 \\ 0 & -2 & 2 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\]

The transfer function is then given by

\[
G(s) = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} s^{-1} + \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix} s^{-2} + \begin{bmatrix} -4 & 0 \\ 0 & -2 \end{bmatrix} s^{-3} + \begin{bmatrix} 0 & -4 \\ 4 & 0 \end{bmatrix} s^{-4} + \begin{bmatrix} 0 & 9 \\ 0 & -8 \end{bmatrix} s^{-6} + \ldots
\]

\[
= \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}
\]

It should be noted that the Markov parameters are computed as follows

\[
y_k = C[A^{k-1}B]
\]

\[
y_{k+1} = C[A[A^{k-1}B]] \quad \forall k \geq 0
\]

Using the algorithm of Section 5.2, for the case \( m \leq n \) gives

\[
g_{11}(s) = \frac{s^2 + 2s + 2}{s^2 + 2}
\]

\[
g_{12}(s) = -g_{21}(s) = \frac{2}{s^2 + 2}
\]
Therefore,

\[
G(s) = \begin{bmatrix}
\frac{s^2 + 2s + 2}{s^2 + 2} & \frac{2}{s^2 + 2} \\
-\frac{2}{s^2 + 2} & \frac{s^3 + 2s + 3s + 2}{s(s^2 + 2)}
\end{bmatrix}
\]

5.4 Criterion for the Existence of the Inverse

The system to be inverted may be described in state-space form, Equation (5.3.1), or by a set of differential equations, Equation (5.3.7), or by a matrix-transfer function of the form

\[
G(s) = \sum_{i=0}^{n} A_i s^i
\]

(5.4.1)

where \( A_i \) (\( i = 0, 1, 2, \ldots, n \)) are constant (\( p \times q \)) matrices, and \( b_i \) (\( i = 0, 1, 2, \ldots, n \)) are constant scalars.

5.4.1 L-Integral Inverses [70, 83]

Definition 5.1

The system \((\hat{A}, \hat{B}, \hat{C}, \hat{D})\) is an L-integral inverse of the system \((A, B, C, D)\) if

\[
\hat{G}(s)G(s) = \frac{1}{s} L I_q
\]

(5.4.2)

where \( I_q \) is the (\( q \times q \)) identity matrix. It should be noted that compatibility in (5.4.2) requires that the matrix dimensions \( \hat{q}, \hat{p} \) and \( p, q \) associated with the inverse must satisfy

\[
\hat{q} = p
\]

\[
\hat{p} = q
\]

According to (5.4.2) \( \hat{G}(s) \) is just a left inverse of \( G(s) \), so that \( p \geq q \) is always a necessary condition for the system to have an L-integral inverse, that is the system must have at least as many
outputs as inputs. It should be noted that \( \hat{n} \) is not defined by 
\( q, n \) and \( p \) alone.

**Definition 5.2**

The system \((A,B,C,D)\) is invertible if it has an \( L \)-integral inverse for some finite \( L \). The least integer \( L \) for which an \( L \)-integral inverse exists is called the inherent integration of an invertible system.

From Equation (5.4.2) it is clear that a system is invertible if and only if the transfer function matrix \( G(s) \) has rank \( q \) over the field of rational functions in \( s \).

### 5.4.2 Test for Invertibility

Necessary and sufficient conditions for the existence of an \( L \)-integral inverse of a system have been given in reference [83]. Here we will simply reproduce them. Define

\[
M_0 = D
\]

\[
M_i = \begin{bmatrix}
D & 0 & 0 & \cdots & 0 \\
CB & D & 0 & \cdots & 0 \\
CAB & CB & D & \cdots & C \\
\vdots & \vdots & \vdots & \cdots & \vdots \\
CA^{i-1}B & CA^{i-2}B & CA^{i-3}B & \cdots & D
\end{bmatrix}
\]  

**Theorem 5.3**

For any non-negative integer \( L \),

\[
\text{rank} \left( M_L \right) - \text{rank} \left( M_{L-1} \right) \leq q
\]  

with equality if and only if the system has an \( L \)-integral inverse.

**Theorem 5.4**

A system is invertible if and only if

\[
\text{rank} \left( M_n \right) - \text{rank} \left( M_{n-1} \right) = q
\]  

where \( n \) is the state-dimension.
It should be noted that rank \( (M_{-1}) \) is defined to be zero.

Thus Theorem 5.3 can be used to compute the value of \( L \), while Theorem 5.4 can be used to test whether a system is invertible or not. In Section 5.5.2 an alternative criterion for the existence of an inverse is given which is computationally very much easier.

### 5.5 Inversion Algorithm

In this Section we will assume that the system has an equal number of inputs and outputs, that is, \( p = q \). In the next Section the algorithm will be generalised to the case when \( p > q \).

From Equation (5.4,2) it is clear that the inverse is given by \( G^{-1}(s) \), and that it exists as long as \( \det(G(s)) \) is not identically zero. We will consider three distinct cases. The first case, is when the first non-zero matrix in the sequence of Markov parameters is non-singular. The second case is when the first non-zero matrix in the time-moments sequence is non-singular. The third case, which is the general case, is when neither of the first two cases can be applied.

#### 5.5.1 The first non-zero Markov parameter is non-singular

It has been shown how a system may be reduced to the following
\[
G(s) = D_0 + D_1 s^{-1} + D_2 s^{-2} + \ldots
\]  

(5.5.1)

Suppose that the first non-zero matrix in the sequence of Markov parameters \( D_0, D_1, D_2, \ldots \), is \( D_k \), and further suppose it is non-singular. Then Equation (5.5.1) may be rewritten as
\[
G(s) = \frac{1}{s^k} \{D_k + D_{k+1} s^{-1} + D_{k+2} s^{-2} + \ldots\} 
\]  

(5.5.2)

\[
= \frac{1}{s^k} \overline{G}(s)
\]  

(5.5.3)

Using Theorem 5.3, and since \( D_k \) is non-singular, it is found that for \( \overline{G}(s) \)

\[
L = 0
\]  

(5.5.4)
Let
\[ G^{-1}(s) = s^k [G(s)]^{-1} \]  \hspace{1cm} (5.5.5)
and let
\[ [G(s)]^{-1} = E_0 + E_1 s^{-1} + E_2 s^{-2} + \ldots \]  \hspace{1cm} (5.5.6)

Hence from the definition of the inverse
\[ [G(s)]^{-1} G(s) = I_q \]  \hspace{1cm} (5.5.7)
where \( I_q \) is the \((qxq)\) identity matrix. Thus
\[ [E_0 + E_1 s^{-1} + E_2 s^{-2} + \ldots][D_k + D_{k+1}s^{-1} + D_{k+2}s^{-2} + \ldots] = I_q \]  \hspace{1cm} (5.5.8)

Hence equating coefficients and assuming \( D_k \) is non-singular we get
\[ E_0 = D_k^{-1} \]
\[ E_1 = -D_k^{-1} D_{k+1} E_0 \]
\[ E_i = -D_k^{-1} \left( \sum_{j=1}^{i-1} D_{k+j} E_{i-j} \right) \] \( \forall i > 0 \) \hspace{1cm} (5.5.9)

Thus using Equation (5.5.9), the Markov parameters of the inverse system, \( E_i \) \( (i = 0, 1, 2, \ldots) \) can be computed very easily.

It should be noted that only one \((qxq)\) constant matrix inversion is required. The inverse can then be written as
\[ [G(s)]^{-1} = s^k [G(s)]^{-1} \]
\[ = [E_0 s^k + E_1 s^{k-1} + \ldots + E_{k-1}s] + [E_k + E_{k+1}s^{-1} + E_{k+2}s^{-2} + \ldots] \]
\[ = E(s) + D(s) \]  \hspace{1cm} (5.5.10)

where
\[ E(s) = \sum_{i=0}^{k-1} E_i s^{k-1} \]
and
\[ D(s) = \sum_{i=0}^{\infty} E_{k+i} s^{-i} \]  \hspace{1cm} (5.5.11)

Thus the inverse system is simply a dynamical system \( D(s) \) coupled with a differential system \( E(s) \) of order \( k \).

Since the system is finite dimensional, then its inverse must also be finite dimensional and therefore the infinite series (5.5.11) must reduce to a regular rational transfer function matrix, that is,
none of the entries of $D(s)$ have a numerator of degree greater than the denominator.

By substituting $z = \frac{1}{s}$ in Equation (5.5.11) the algorithm of Section 5.2 (for the case $m \leq n$) may be used to compute the individual entries of $D(s)$. The inverse matrix transfer function is then given by Equation (5.5.10).

A state-vector representation of the inverse system may be obtained by applying any one of the many algorithms for the minimal realization of systems [43, 77, 89].

5.5.2 The first non-zero time-moment is non-singular

If the assumptions in Section 5.5.1 are not satisfied then the system is reduced to the following form

$$G(s) = C_0 + C_1 s + C_2 s^2 + \ldots$$

(5.5.12)

where the $C_i$ ($i = 0, 1, 2, \ldots$) are constant $(q \times q)$ matrices, which have been shown to be directly proportional to the time-moments of the system. Suppose that the first non-zero matrix in the sequence $C_0', C_1', C_2', \ldots$, is non-singular and let it be $C_k$. Let

$$G(s) = s^k [C_k + C_{k+1} s + C_{k+2} s^2 + \ldots]$$

(5.5.13)

$$= s^k G^*(s)$$

(5.5.14)

Let

$$G^{-1}(s) = s^{-k} [G^*(s)]^{-1}$$

(5.5.15)

Let

$$[G^*(s)]^{-1} = F_0 + F_1 s + F_2 s^2 + \ldots$$

(5.5.16)

Then by the definition of the inverse

$$G^{-1}(s) G(s) = [G^*(s)]^{-1} G^*(s) = I_q$$

(5.5.17)

where $I_q$ is the $(q \times q)$ identity matrix. Thus

$$[F_0 + F_1 s + F_2 s^2 + \ldots][C_k + C_{k+1} s + C_{k+2} s^2 + \ldots] = I_q$$

(5.5.18)

Equating coefficients and assuming $C_k$ is non-singular gives

$$F_0 = C_k^{-1}$$
\[ F_1 = -C_k^{-1} C_{k+1} F_0 \]
\[ F_2 = -C_k^{-1} [C_{k+1} F_1 + C_k F_0] \]
and in general
\[ F_i = -C_k^{-1} \sum_{j=1}^{i} C_{k+j} F_{i-j} \quad \forall i > 0 \]  

(5.5.19)

Thus using (5.5.19) the time-moments of the inverse system, \( F_i \) (\( i = 0, 1, 2, \ldots \)) may be computed very easily. Again, only one constant matrix inversion is required, but it should be noted that (5.5.19) is of the same form as (5.5.9).

Since the system is finite dimensional, then its inverse must also be of finite dimensions and therefore the series expansion (5.5.16) is equivalent to a rational matrix transfer function. However in this case it is unknown what the order of the numerator is with respect to that of the denominator. Hence the algorithm of Section 5.2 (for the general case \( m > n \)) may be used to compute the individual entries of \([G^*(s)]^{-1}\). The inverse matrix transfer function is then given by (5.5.15).

5.5.3 The general case

The algorithm introduced in this Section computes the inverse, if it exists, without having to determine the value of \( L \) (Theorem 4.1).

Suppose that the system transfer function \( G(s) \) is given. Further suppose that \( D_0 \) (Equation 5.5.1) is non-zero and that it is singular. The following substitution for \( s \) is made in \( G(s) \)
\[ s = \frac{1 + \alpha z}{z} \]  
(5.5.20)

\( G(z) \) is then expanded into a power series expansion, about \( z = \infty \), to give
\[ G(z) = D_0 + D_1 z^{-1} + D_2 z^{-2} + \ldots \]  
(5.5.21)

where \( D_i \) (\( i = 0, 1, 2, \ldots \)) are constant \((q \times q)\) matrices. The constant \( \alpha \) is chosen such that \( D_0 \) in (5.5.21) is non-singular.
Theorem 5.5

If the given system is invertible, then there exists an α such that \( D_0 \) in Equation (5.5.21) is non-singular.

Proof

The proof is very simple. Noting that \( D_0 \) in (5.5.21) is given by
\[
D_0 = G(s) \bigg|_{s=\alpha}
\]  
(5.5.22)

By definition the system is invertible if and only if \( \det[G(s)] \) is not identically zero. Hence if the system is invertible then
\[
\det[G(\alpha)] \neq 0 \text{ for some } \alpha
\]  
(5.5.23)

therefore
\[
\det D_0 = \det[G(\alpha)] \neq 0 \text{ for some } \alpha
\]  
(5.5.24)

Care should be taken when choosing \( \alpha \) to ensure that it is not a pole of \( G(s) \).

A necessary and sufficient condition for the existence of an inverse is given by the following theorem.

Theorem 5.6

If a system is of dimension \( n \), then if after substituting \( (n+1) \) different values of \( \alpha \) (\( \alpha \) not a pole of the system) Equation (5.5.24) is not satisfied, then the system is not invertible.

Proof

Consider the matrix transfer function of a proper system (that is, the numerator of any entry of \( G(s) \) is of degree not greater than the denominator). Then it is easy to show
\[
\det[G(s)] = \frac{n(s)}{d(s)}
\]  
(5.5.25)

where \( n(s) \) and \( d(s) \) are polynomials in \( s \). The polynomial \( n(s) \) is of degree less than, or equal to, that of \( d(s) \).

Further the dimension of the system, \( n \), is greater than, or
equal to, the degree of \( d(s) \). Hence \( n \) is greater than, or equal to, the degree of \( n(s) \). Also

\[
\det[G(s)] = 0
\]

if, and only if, \( n(s) \) is identically zero. Hence if

\[
\begin{align*}
\det G(\alpha_i) &= 0 \quad (5.5.26) \\
n(\alpha_i) &= 0 \quad (5.5.27)
\end{align*}
\]

that is, \( \alpha_i \) is a zero of \( n(s) \).

Thus if

\[
\det[G(\alpha_i)] = 0 \quad i = 1, 2, \ldots, n, n+1
\]

and

\[
\alpha_i \neq \alpha_j \quad \forall i \neq j
\]

Then \( n(s) \) must be identically zero. Hence \( \det G(s) = 0 \).

Therefore the system is not invertible.

Theorem 5.6 can be seen to be a test for the invertibility of systems. The amount of computation required is to determine the rank of \((n+1)(qxq)\) constant matrices while the test given in Theorem 5.4 requires the determination of the rank of an \((nqxnxq)\) and \( [(n-1)qx(n-1)q] \) constant matrices.

For stable systems an ideal choice for \( \alpha \) would be to take positive values.

The equivalence of the above analysis in state-space can be made as follows.

\[
G(s) = C(sI-A)^{-1}B + D
\]

Therefore

\[
\begin{align*}
G(z) &= C\left(\frac{I+azI}{z} - A\right)^{-1}B + D \\
&= zC(I-(A-\alpha I)z)^{-1}B + D \\
&= -zC(A-\alpha I)^{-1}[zI-(A-\alpha I)^{-1}]^{-1}B + D \\
&= -[C(A-\alpha I)^{-1}B-D] - C(A-\alpha I)^{-2}Bz^{-1} - C(A-\alpha I)^{-3}Bz^{-2} + \ldots
\end{align*}
\]

\[(5.5.29)\]

Hence
\[
\text{det} D_O = \text{det}[D - C(A-\alpha I)^{-1} B] \tag{5.5.30}
\]

Thus, \(\alpha\) is chosen such that \(\text{det} D_O\), given by (5.5.30) is non-singular.

If an \(\alpha\) is found such that Equation (5.5.24), or Equation (5.5.30) is satisfied, then the procedure of Section 5.5.1 is carried out to determine the inverse of \(G(z)\), that is, to find \([G(z)]^{-1}\).

Then by making the substitution
\[
z = \frac{1}{s-\alpha} \tag{5.5.31}
\]
in \([G(z)]^{-1}\), the system inverse \([G(s)]^{-1}\) is obtained.

In this Section, that is for the general case, we have used the Markov parameters of the system for the inversion process. It can be easily shown that the time-moments of the system may be used, and in that case the substitution required is of the form
\[
s = z + \alpha \tag{5.5.32}
\]

**Example 2**

This example was chosen to illustrate how the inversion algorithm may be used.

\[
G(s) = \begin{bmatrix}
\frac{1}{(s+1)} & \frac{2}{(s+1)} \\
-1 & 0 \\
(s+1)(s+2) & (s+2)
\end{bmatrix}
\]

Expanding \(G(s)\) about \(s = \infty\), gives
\[
G(s) = D_0 + D_1 s^{-1} + D_2 s^{-2} + \ldots
\]

where
\[
D_0 = \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix}, \quad D_1 = \begin{bmatrix}
1 & 2 \\
0 & 1
\end{bmatrix}, \quad D_2 = \begin{bmatrix}
-1 & -2 \\
-1 & -2
\end{bmatrix}, \quad D_3 = \begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix}, \quad D_4 = \begin{bmatrix}
-1 & -2 \\
-7 & -8
\end{bmatrix}, \quad D_5 = \begin{bmatrix}
15 & 16
\end{bmatrix}
\]

... etc.

Let
\[
[G(s)]^{-1} = s[G(s)]^{-1}
\]
where

\[ [G(s)]^{-1} = E_0 + E_1 s^{-1} + E_2 s^{-2} + \ldots \]

then using Equation (5.5.9) we get:

\[
E_0 = \begin{bmatrix} 1 & -2 \\ 0 & 1 \end{bmatrix}, \quad E_1 = \begin{bmatrix} -1 & 0 \\ 1 & 0 \end{bmatrix}, \quad E_2 = \begin{bmatrix} 4 & -4 \\ -2 & 2 \end{bmatrix}, \quad E_3 = \begin{bmatrix} -12 & 12 \\ 6 & -6 \end{bmatrix}, \quad E_4 = \begin{bmatrix} 36 & -36 \\ -18 & 18 \end{bmatrix}, \quad \ldots \text{ etc.}
\]

Therefore

\[ [G(s)]^{-1} = E_0 s + E_1 + E_2 s^{-1} + E_3 s^{-2} + \ldots \]

or

\[ [G(s)]^{-1} = E_0 s + D(s) \]

Then applying the algorithm of Section 5.2 on \( D(s) \) we get

\[
[G(s)]^{-1} = \begin{bmatrix} s & -2s \\ 0 & s \end{bmatrix} + \begin{bmatrix} -\frac{s+1}{s+3} & -4 \\ \frac{s+1}{s+3} & 2 \end{bmatrix} = \begin{bmatrix} \frac{s^2+2s+1}{s+3} & -\frac{4s+2s^2}{s+3} \\ \frac{s+1}{s+3} & \frac{2s+3s+2}{s+3} \end{bmatrix}
\]

It should be noted that a state-space representation of \( D(s) \) may be obtained by using any of the algorithms of references [43, 77, 89].

**Example 3**

This example has been chosen to show how the substitution procedure may be used.

\[
G(s) = \begin{bmatrix} \frac{1}{(s+2)} & 1 \\ s+3 & \frac{1}{(s+1)} \end{bmatrix}
\]

Expanding \( G(s) \) about \( s = \infty \) gives

\[
D_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}
\]

which is singular.

Thus let

\[ s = \frac{z+1}{z}, \quad \text{i.e. } \alpha = 1. \]
Expanding \( G(z) \) about \( z = \infty \) gives

\[
G(z) = D_0 + D_1 z^{-1} + D_2 z^{-2} + \ldots
\]

where

\[
D_0 = \begin{bmatrix}
1/3 & 1/2 \\
2/3 & 1/2
\end{bmatrix}
\]

and \( \det D_0 \neq 0 \)

\[
D_1 = \begin{bmatrix}
-1/9 & -1/4 \\
-7/18 & -1/4
\end{bmatrix}, \quad D_2 = \begin{bmatrix}
1/27 & 1/8 \\
23/108 & 1/8
\end{bmatrix}
\]

\[
D_3 = \begin{bmatrix}
-1/81 & -1/16 \\
-73/648 & -1/16
\end{bmatrix}
\]

Then

\[
\left\{ G(z) \right\}^{-1} = \begin{bmatrix}
\frac{(6z^2 + 5z + 1)}{2z^2} & \frac{(6z^2 + 5z + 1)}{2z^2} \\
\frac{8z^2 + 6z + 1}{2z^2} & \frac{(4z^2 + 4z + 1)}{2z^2}
\end{bmatrix}
\]

Then substituting \( z = \frac{1}{s-1} \), gives

\[
\left\{ G(s) \right\}^{-1} = \begin{bmatrix}
-(0.5s^2 + 1.5s + 1) & (0.5s^2 + 1.5s + 1) \\
(0.5s^2 + 2s + 1.5) & -(0.5s^2 + s + 0.5)
\end{bmatrix}
\]

It should be noted that for this example, the first time-moment is non-singular, and thus the algorithm of Section 5.5.2 may be used. However our purpose was to show how the substitution method may be used.

5.6 Generalised Inverse

In this Section the algorithm of Section 5.5 is generalised for the case when the system to be inverted has more outputs than inputs, that is, \( q < p \).

It has been shown how a system may be expanded in the form

\[
G(s) = D_0 + D_1 s^{-1} + D_2 s^{-2} + \ldots
\]  \hspace{1cm} (5.6.1)

where the \( D_i \) \((i = 0, 1, 2, \ldots)\) are \((p \times q)\) constant matrices.
From Equation (5.4.2) it can be seen that the system is invertible if, and only if, the transfer function matrix $G(s)$ has rank $q$ over the field of rational functions in $s$. It should be noted [70] that the inverse of the system (5.4.1) is non-unique for the case $q < p$.

If the system is invertible, then by rearranging the rows of $G(s)$ we can write

$$G(s) = \begin{bmatrix} G_1(s) \\ G_2(s) \end{bmatrix}$$

where $G_1(s)$ is a $(qxq)$ rational matrix and is of full rank. $G_2(s)$ is a $[(p-q)xq]$ rational matrix. Thus $\det[G_1(s)]$ is not identically zero.

The procedure of Section 5.5 is then used to find the inverse of $G_1(s)$, that is, $[G_1(s)]^{-1}$.

An inverse of $G(s)$ is then given by

$$[G(s)]^{-1} = [[G_1(s)]^{-1}, \mathbf{0}]$$

where $\mathbf{0}$ is the $[(p-q)xq]$ zero matrix.

It should be noted that

$$[G(s)]^{-1}G(s) = [[G_1(s)]^{-1}, \mathbf{0}] \begin{bmatrix} G_1(s) \\ G_2(s) \end{bmatrix} = [G_1(s)]^{-1}G_1(s) = I_q$$

In physical terms, the method simply uses $q$ linearly independent outputs to determine the $q$ linearly independent inputs.

Thus the inversion algorithm for a general multivariable system may be summarised as follows:

I. The system is expanded into a power series about $s = \infty$;

II. The first non-zero Markov parameter is tested to see whether it is of rank $q$. If it is go to VI;

III. The system is expanded into a power series about $s = 0$;

IV. The first non-zero time-moment is tested to see whether it is of rank $q$. If it is go to VI;

V. Make the substitution $s = \frac{1+\alpha z}{z}$, and test whether the first non-zero Markov parameter of $G(\hat{z})$ is of rank $q$. If it is then go to VI. If it is not then another substitution is made with a different value of $\alpha$. If after substituting $(n+1)$ different $\alpha$'s it is found that the first Markov parameter of $G(\hat{z})$ is of rank...
less than \( q \), then the system is not invertible, and computation stops;

VI. Taking any \( q \)-linearly independent rows of \( G(z) \), chosen to correspond with \( q \)-linearly independent rows of the first Markov parameter (time-moment), \( G_1(z) \) is formed;

VII. The Markov parameters (time-moments) of \( G_1(z)^{-1} \) are computed using Equation (5.5.9) (Equation (5.5.19));

VIII. Algorithm of Section 5.2 is used to compute the individual entries of \( [G_1(z)]^{-1} \). Then substituting \( z = \frac{1}{s-\alpha} \) to get \( [G_1(s)]^{-1} \) (only if substitution was made in the first place);

IX. The left inverse of the system is given by \( [G(s)]^{-1} = [[G_1(s)]^{-1}, 0] \).

For the special case when \( p = q \), we have \( [G(s)]^{-1} = [G_1(s)]^{-1} \).

Example 4

Consider the following system [70]

\[
G(s) = \begin{bmatrix}
\frac{1}{s+2} & \frac{1}{s+1} \\
\frac{s+3}{(s+1)(s+2)} & \frac{s}{(s+1)} \\
\frac{s(s+3)}{(s+1)(s+2)} & 0
\end{bmatrix}
\]

Expanding \( G(s) \) about \( s = \infty \) would give

\[
D_0 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}
\]

which is seen to be of rank 2. In fact row 2 and row 3 are the linearly independent rows. Hence take

\[
G_1(s) = \begin{bmatrix}
\frac{s+3}{(s+1)(s+2)} & \frac{s}{(s+1)} \\
\frac{s(s+3)}{(s+1)(s+2)} & 0
\end{bmatrix}
\]

and carrying out the inversion algorithm we get

\[
[G_1(s)]^{-1} = \begin{bmatrix} 0 & \frac{(s+1)(s+2)}{s(s+3)} \\ \frac{s+1}{s} & -\frac{s+1}{s} \end{bmatrix}
\]
It should be noted that this is not a unique inverse.

The advantages of this algorithm over other algorithms may be listed as follows:

1. It is conceptually very simple;
2. It is computationally simple;
3. A separate test for the existence of the inverse is unnecessary;
4. It can be applied to systems described in state-space form or by a matrix transfer function or by a set of constant differential equations;
5. The inverse system may be obtained as a matrix transfer function or in state-space form.
Chapter 6

MINIMAL REALIZATION THEORY AND THE
MODEL REDUCTION PROBLEM

6.1 Introduction

In Chapters 3 and 4, continued fraction methods and Padé approximation techniques were used for reducing the order of linear systems. The reduced models were in transfer function form. In this Chapter we will outline the relationship between the seemingly different methods of reduction and the theory of minimal realizations, and use the latter to derive equivalent reduced order models described in state-space form.

In Section 6.2, the problem of minimal realization of linear time-invariant systems is discussed. It is shown how a system may be realized from its Markov parameters, time-moments, or a combination of both. Hence reduced order models, described by a state-vector representation may be derived directly from the time-moments and Markov parameters of the system.

In Chapter 4, the continued fraction methods and the Padé approximation techniques were extended to the problem of reducing the order of multivariable systems. The reduced order models were described by a set of constant differential equations. Thus in Section 6.3 an algorithm is presented for the realization of systems described by a set of constant differential equations. The algorithm is a modification of existing minimal realization algorithms based on the input-output measurements of the system [1, 43, 77, 88].

In Section 6.4, the problem of minimal partial realizability which is the problem of realizing a finite number of Markov parameters, first introduced by Tether [93], and its relation to the problem of model reduction is given. There have been a number of algorithms for the
solution of the minimal partial realization problem [48, 77, 93],
but unfortunately the solutions may be non-unique, due to the fact
that the given finite sequence of parameters is not sufficient to
give an exact solution of the realization problem. However, in the
problem of model reduction, the non-uniqueness problem can be
eliminated by simply making the reduced model 'fit' more of the
parameters of the original system.

Hence in Section 6.5, Silverman's algorithm for the minimal
realization of an infinite sequence of parameters is described, and
then together with the theory developed in Section 6.2, the algorithm
is used to illustrate how reduced order models may be derived by the
simple partitioning of characteristic matrices of the original
system. In fact any of the algorithms [1, 43, 77, 89] may be used
for the reduction process, but the algorithm of Rissanen [77] seems
to be computationally the simplest. Hence it follows that if a
minimal realization algorithm is available, then it will be unnecessary
to write a new model reduction algorithm. For single-input-single-output
systems the models derived in this section are equivalent to those
derived in Chapter 3. But for multivariable systems, the reduced
models may not necessarily be the same as those of Chapter 4. It
should also be noted that in this section our purpose is to use the
minimal realization algorithms rather than to discuss individual
algorithms. Two examples are given to show how Silverman's algorithm
may be applied to the reduction of multivariable systems.

6.2 Minimal Realization of Linear Dynamical Systems

In this Chapter, we will be mainly concerned with constant
continuous-time linear system representations of the form
\[ \dot{x} = Ax + Bu \]
\[ y = Cx \]  
(6.2.1)
where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^q$ and $y \in \mathbb{R}^p$ are the state, input and output vectors respectively. $A$, $B$, and $C$ are constant matrices of orders compatible with $x$, $u$ and $y$. A system representation of this type will be denoted by $(A, B, C)$.

A second description of a constant linear system is given by its transfer function matrix

$$G(s) = C(sI-A)^{-1}B$$

(6.2.2)

It can be shown that $G(s)$ is a function that maps $u(t) \rightarrow y(t)$. Clearly $(A, B, C)$ give rise to only one transfer function matrix, but the converse is not generally true. The realization problem may be stated as: given $G(s)$ find a triple $(A, B, C)$ such that these three matrices realize $G(s)$. A minimal realization is one in which the dimension of the state matrix $A$ is minimal.

A realization is of minimum dimension if it is completely controllable and completely observable. This necessary and sufficient condition was presented and proved by Kalman [47], and has been an important axiom in several algorithms devised to obtain minimal realizations.

The first two algorithms were presented simultaneously by Kalman [47] and Gilbert [38]. Gilbert proposed a sequence of linear algebraic computations which mapped the transfer function into a state vector representation. His algorithm placed a lot of emphasis on the residue matrices of the poles of the transfer function matrix.

In Kalman's algorithm, a different approach was adopted. The construction of a minimal realization is divided into two steps: the formation of a controllable (or observable) realization and the subsequent reduction of this realization to minimum dimensions. The essential elements of the procedure proposed by Kalman for the first step have been incorporated into algorithms devised later by Wolovich and Falb [102], and others for the realization problem.
Many algorithms have been developed for the minimal realization of a system described by a record of its input-output data, namely the Markov parameters of the system [1, 43, 77, 89]. Of these algorithms, the most widely used is that by Ho and Kalman [43]. However for the purpose of model reduction we have found that the algorithms of Silverman [89] and Rissanen [77] for the minimal realization of systems are most suitable. A survey of minimal realization algorithms has been made by Roessen [81].

6.2.1 Realizability in terms of the Markov parameters

Given the system of Equation (6.2.2), we can expand G(s) in a negative power series as follows:

\[
G(s) = CBs^{-1} + CABs^{-2} + CA^2Bs^{-3} + \ldots \quad (6.2.3)
\]

\[
= D_1s^{-1} + D_2s^{-2} + D_3s^{-3} + \ldots \quad (6.2.4)
\]

where

\[
D_i = CA^{i-1}B, \quad i = 1, 2, 3, \ldots \quad (6.2.5)
\]

\[\Delta\] the Markov parameters of the system.

Thus given the infinite series (6.2.4), the problem is that of determining whether it is realizable. If it is realizable then any of the algorithms of references [1, 43, 77, 88] may be used to compute a minimal realization.

Define a sequence of Hankel matrices for the system as

\[
H_{ij} = \begin{bmatrix}
D_1 & D_2 & \ldots & D_j \\
D_2 & D_3 & \ldots & D_{j+1} \\
\vdots & \vdots & \ddots & \vdots \\
D_i & D_{i+1} & \ldots & D_{i+j-1}
\end{bmatrix} \quad (6.2.6)
\]

Theorem 6.1 [43, 89]

(a) An infinite sequence, (6.2.4), is realizable if and only if there exist non-negative integers \(\alpha, \beta,\) and \(n\) such that for 
\[j = 1, 2 \ldots\]
rank \( H_{\beta,\alpha} = rank H_{\beta+1,\alpha+j} = n \) \hspace{1cm} (6.2.7)

(b) If (6.2.4) is realizable then \( n \) as defined in (a) is the dimension of the minimal realization.

Note that if (6.2.4) is generated by a triple \((A,B,C)\) with controllability and observability matrices \( Q_{\alpha} \) and \( R_{\beta} \), then

\[
H_{\beta,\alpha} = R_{\beta}Q_{\alpha}^{-1}
\]

\[
\begin{bmatrix}
C \\
CA \\
CA^2 \\
\vdots \\
CA^{\beta-1}
\end{bmatrix} = \begin{bmatrix}
B, AB, A^2B, \ldots, A^{\alpha-1}B
\end{bmatrix}
\]

6.2.2 Realizability in terms of time-moments

Assuming that the system is asymptotically stable and hence that \( A \) is non-singular, the matrix transfer function may be expanded as follows

\[
G(s) = -CA^{-1}[I+A^{-1}s+A^{-2}s^2+\ldots]B
\]

\[
= -CA^{-1}B - CA^{-2}Bs - CA^{-3}Bs^2 - \ldots
\]

\[
= -C_1 - C_2s - C_3s^2 - \ldots
\]

where

\[
C_i = CA^{-i}B \quad i = 1, 2, 3, \ldots \hspace{1cm} (6.2.11)
\]

\[
= (-1)^{(i+1)} \frac{(i+1)}{(i)!} x (i \text{ th time-moment of the system}).
\]

To determine whether the infinite series (6.2.10) is realizable, define the following sequence of Hankel matrices for the system

\[
H'_{ij} = \begin{bmatrix}
C_1 & C_2 & \ldots & C_j \\
C_2 & C_3 & \ldots & C_{j+1} \\
\vdots & \vdots & \ddots & \vdots \\
C_i & C_{i+1} & \ldots & C_{i+j-1}
\end{bmatrix} = \begin{bmatrix}
CA^{-1} \\
CA^{-2} \\
\vdots \\
CA^{-\beta}
\end{bmatrix} \begin{bmatrix}
B, A^{-1}B, \ldots, A^{-1+j}B
\end{bmatrix}
\]

\[
(6.2.12)
\]
Then

\[ \text{rank } H'_{ij} = \text{rank } [R_i^j Q_i^j] = \text{rank } [R_i^1 A^{-1-i-j} Q_j^j] = A[R_i^1 A^{-1-i-j} Q_j^j] \]

and for a minimal realization,

\[ \text{rank } R_{\beta} = \text{rank } Q_{\alpha} = \text{rank } A \]

\[ \text{rank } H'_{\beta,\alpha} = \text{rank } H_{\beta,\alpha} \] (6.2.13)

Hence using (6.2.12), Theorem 6.1 may be applied to determine the minimal dimension for the system described by (6.2.10) [17]. If the three matrices \((A^{-1}, B, CA^{-1})\) are taken as being unknown, then any of the algorithms of references [1, 43, 77, 89] can be applied without modification. At the end of the calculation it is necessary, of course, to invert the matrix \(A^{-1}\) to obtain \(A\). The realization so constructed is minimal as seen from (6.2.13).

6.2.3. Realizability in terms of time-moments and Markov parameters

Again the system is assumed to be asymptotically stable, and hence that \(A\) is non-singular. We would like to obtain a minimal realization using the first \(k\) Markov parameters, and the minimum number of time-moments (say). This is useful in identification when only the first \(k\) Markov parameters can be measured accurately [17, 81, 82].

The system matrix transfer function may be expanded as follows

\[ G(s) = D_1 s^{-1} + D_2 s^{-2} + D_3 s^{-3} + ... D_k s^{-k} + ... \] (6.2.14)

\[ G(s) = -C_1 + C_2 s^{-1} - C_3 s^{-2} + ... \] (6.2.15)

where the \(D_i\) and \(C_i\) are as defined in (6.2.5) and (6.2.11).

To determine whether the system defined by (6.2.14) as far as \(D_k\) and (6.2.15) is realizable, define the following sequence of Hankel matrices.
then we have

\[
H''_{ij} = \begin{bmatrix}
D_k & D_{k-1} & \cdots & D_1 & C_1 & C_2 & \cdots & C_{j-k} \\
D_{k-1} & D_{k-2} & \cdots & C_1 & C_2 & C_3 & \cdots & C_{j-k+1} \\
& \ddots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
D_1 & C_1 & \cdots & \cdots & \cdots & \cdots & C_{j-1} \\
C_1 & C_2 & \cdots & \cdots & \cdots & \cdots & \cdots \\
& \ddots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
C_{i-k} & C_{i-k+1} & \cdots & \cdots & \cdots & \cdots & C_{i+j-k+1}
\end{bmatrix}
\] (6.2.16)

and for a minimal realization

\[
rank H''_{ij} = rank [R''_{i,j}] = rank [R_{i,j}^{-1}] = rank H_{i,j}
\]

and for a minimal realization

\[
rank H''_{i,j} = rank (R_{i,j} Q) = rank H_{i,j}
\]

Hence using (6.2.16), Theorem 6.1 may be applied to
determine the minimal dimension for the system described by the
sequence

\[
[D_k \quad \cdots \quad D_1 \quad C_1 \quad C_2 \quad \cdots]
\] (6.2.18)

If any of the algorithms of references [1, 43, 77, 89] are
applied to (6.2.18), a minimal realization \((F,G,H)\) is obtained where

\[
F = A^{-1} \\
G = B \\
H = CA^{-k}
\]
6.3 Realization of a Set of Constant Differential Equations

So far, we have assumed that the system is given in the form of (6.2.1) or (6.2.2). However, it may be necessary to construct a state-space realization from an external input-output relationship comprising a set of constant differential equations [37, 101] of the form

\[ L(D) \dot{z} = M(D)u, \quad D = \frac{d}{dt} \] \hspace{1cm} (6.3.1)

where \( L(D) \) and \( M(D) \) are \((p \times p)\) and \((p \times q)\) polynomial matrices.

Equation (6.3.1) may be rewritten in the form

\[ y(s) = L^{-1}(s)M(s)u(s) \] \hspace{1cm} (6.3.2)

where \( M(s) \) and \( L(s) \) are of the form

\[ M(s) = M_0 + M_1s + M_2s^2 + \ldots + M_{n-1}s^{n-1} \] \hspace{1cm} (6.3.3)

\[ L(s) = L_0 + L_1s + L_2s^2 + \ldots + L_n s^n \] \hspace{1cm} (6.3.4)

with \( M_i, L_i \) \((i = 0, 1, \ldots)\) being \((p \times q)\) and \((p \times p)\) constant matrices respectively. The matrix \( L(s) \) is assumed to be invertible.

This form of description was obtained when using Padé approximation methods for the reduction of multivariable systems, in Chapter 4.

Some methods for deriving a state-variable model for the system of equation (6.3.1) have been given [37, 101]. In this section our purpose will be to derive a state-vector representation using existing algorithms for the realization of a system described by a matrix transfer function.

If the matrix transfer function \( G(s) \), given by

\[ G(s) = L^{-1}(s)M(s) \] \hspace{1cm} (6.3.5)

can be expanded into a power series expansion (about \( s=0 \) or \( s=\infty \)), then using the theory of Section 6.2, any of the algorithms [1, 43, 77, 89] can be used to realize the system. There are three cases to be considered. The first is when \( L_n \) is non-singular. The second is when \( L_0 \) is non-singular. The third, and this is the general case, is when neither the first two cases hold.
6.3.1 Expansion of the system to get the Markov parameters

Let

\[ G(s) = \left[ \sum_{i=0}^{n} L_i s^i \right]^{-1} \left[ \sum_{i=0}^{n-1} M_i s^i \right] = D_1 s^{-1} + D_2 s^{-2} + D_3 s^{-3} + \ldots \]  \hspace{1cm} (6.3.6)

where the \( D_i \) \((i = 1, 2, \ldots)\) are the Markov parameters of the system. Equating coefficients and assuming \( L_n \) is non-singular, we get the following set of recursive linear matrix equations:

\[ D_1 = L_n^{-1} M_{n-1} \]
\[ D_2 = -L_n^{-1} \left[ L_{n-1} D_1 - M_{n-2} \right] \]

and in general

\[ D_k = L_n^{-1} \left[ M_{n-k} - \sum_{j=1}^{k-1} L^{-1}_{n-j} D_{k-j} \right] \] \hspace{1cm} (6.3.7)

\( \forall k \geq 1, \text{ and } L_i, M_i = 0 \hspace{1mm} \forall i < 0 \)

Hence (6.3.7) can be used to compute all the necessary \( D_i \)'s and any of the algorithms of references [1, 43, 77, 89] may be used to get a minimal realization.

6.3.2 Expansion of the system to get the time-moments

If it is found that \( L_n \) is singular, then we test whether \( L_0 \) is singular. Assuming that \( L_0 \) is non-singular, then let

\[ G(s) = \left[ \sum_{i=0}^{n} L_i s^i \right]^{-1} \left[ \sum_{i=0}^{n-1} M_i s^i \right] = -C_1 - C_2 s - C_3 s^2 - \ldots \]  \hspace{1cm} (6.3.2)

where the \( C_i \) \((i = 1, 2, \ldots)\) are the time-moments of the system. Equating coefficients and assuming \( L_0 \) is non-singular, we get the following set of recursive linear matrix equations

\[ C_1 = -L_0^{-1} M_0 \]
\[ C_2 = -L_0^{-1} \left[ M_1 - L_1 C_1 \right] \]

and in general

\[ C_k = -L_0^{-1} \left[ M_{k-1} - \sum_{j=1}^{k-1} L^{-1}_{j} C_{k-j} \right] \] \hspace{1mm} \forall k > 0 \hspace{1cm} (6.3.9)
Hence (6.3.9) can be used to compute the necessary number of \( C_i \)'s. Then using the method outlined in Section 6.2.2, a triple \((F,G,H)\) is found for the sequence of time-moments. Hence the system may be described by

\[
\dot{x} = A x + B u \\
\gamma = C x
\]

where

\[
A = F^{-1} \\
B = G \\
C = HF^{-1}
\]

and this assumes that \( A \) is non-singular.

So far it has been assumed that either \( L_0 \) or \( L_n \) is non-singular. If this is not the case, then a general procedure is given in the next section.

6.3.3 The general case

If both the matrices \( L_0 \) and \( L_n \) are singular, then we make the following substitution in \( G(s) \),

\[
s = z + \alpha
\]

(6.3.10)

giving \( G(z) \). \( G(z) \) is then expanded into a power series expansion about \( z = 0 \) to give

\[
G(z) = [L(z)]^{-1} M(z) = -E_1 - E_2 z - E_3 z^2 - \ldots
\]

(6.3.11)

where the \( E_i \) (\( i = 1, 2, \ldots \)) are the time-moments of \( G(z) \). The constant \( \alpha \) is chosen such that \( L' \) in (6.3.12) is non-singular,

\[
L(z) = \sum_{i=0}^{n} L'_i z^i = L'_0 + L'_1 z + L'_2 z^2 + \ldots + L'_n z^n
\]

(6.3.12)

we then have the following theorem.

**Theorem 6.2**

If \( L(s) \) is non-singular, then there exist an \( \alpha \) such that \( L'_0 \) is non-singular.
Proof

By definition $L^{-1}(s)$ must exist. Also $L_0'$ in Equation (6.3.12) is given by

$$L_0' = L(s) \bigg|_{s = a}$$

(6.3.13)

and since $L(s)$ is invertible, then $\det[L(s)]$ is not identically zero. Hence

$$\det[L(a)] \neq 0 \text{ for some } a$$

(6.3.14)

Therefore

$$\det L_0' = \det[L(a)] \neq 0 \text{ for some } a$$

(6.3.15)

Care should be taken when choosing $a$, to ensure that it is not a zero of $L(s)$.

Having obtained such an $a$, the coefficients of (6.3.11) are computed using the recursive Equation (6.3.9). Then a triple $[F, G, H]$ is found to realize the sequence $E_1', E_2', E_3', \ldots$, of Equation (6.3.11), minimally using the method of Section (5.2.2).

Hence the state-vector representation of the system is given by

$$A = [F^{-1} + aI]$$

$$B = G$$

$$C = HF^{-1}$$

Hence it is easy to see that $a$ must not be a pole of the system.

6.4 Minimal Partial Realization and its Relation to the Reduction Problem

The minimal realization theory discussed previously dealt with linear, finite-dimensional, constant dynamical systems for which either the transfer function or the infinite sequence of Markov parameters is known exactly. The Ho and Kalman algorithm, for example, requires
that $2r$ Markov parameters be specified exactly (where $r$ is the degree of the annihilating polynomial of the minimal state matrix).

Consider, however, a system which is not finite dimensional but rather is a distributed parameter system. In this case, any finite sequence of parameters $\{D_1, D_2, \ldots, D_N\}$ would provide insufficient data for the solution to the problem. Similarly it may be that the system is finite dimensional, but too few parameters are available for an exact solution, for any of the algorithms [1, 43, 89] to be used. The concept of minimal partial realizability is fairly recent and only a few papers have been written about it so far [36, 48, 77, 82, 93]. The problem may be stated as follows:

Given a finite sequence of Markov parameters, $N_0$ say, available initially, is it possible to find a finite dimensional linear constant system whose first $N_0$ Markov parameters are equal to the given finite sequence? This is called the partial realizability problem. The determination of realizations with the smallest state-space dimension whose first $N_0$ Markov parameters are equal to the given sequence is called the minimal partial realizability problem.

The given initial sequence may not be Markov parameters, but rather time-moments or a combination of both (see Section (6.2)). Thus if the Markov parameters were replaced by the time-moments then the partial realizability problem becomes the Padé approximation problem (see Appendix II); and if $N_0$ consists of a combination of Markov parameters and time-moments, then the partial realizability problem becomes the problem of Padé approximation about two (or more) points. Thus it is obvious that the problem of partial realization and the model reduction problem are, in a sense, equivalent.

Consider the following sequence of Hankel matrices
where $i + j \geq N_0 + 1$.

Suppose that there exist no non-negative integers $\beta$, $\alpha$ such that for $j = 1, 2, \ldots$

$$\text{rank } H_{\beta, \alpha} = \text{rank } H_{\beta+1, \alpha+j} = n;$$

i.e. Theorem 6.1 cannot be applied. Thus a way must be found to extend the given sequence of Markov parameters so that a minimal realization may be obtained.

Tether [93] and Kalman [48] have given a bound on the dimension of the minimal partial realization of a given sequence of parameters. The lemma will be stated here, and the proof may be found in references [48, 93].

**Lemma 6.1**

Let $\{D_1, D_2, \ldots, D_{N_0}\}$ be a finite sequence of Markov parameters and let $J$ be a partial realization whose first $N_0$ Markov parameters are equal to the given sequence. Then the dimension of a minimal partial realization, $\text{min.dim. } J$, satisfies the following inequality

$$\text{min.dim. } J \geq n(N_0) - \rho H_{1, N_0} + \rho H_{2, N_0-1} - \rho H_{1, N_0-1} + \cdots$$

$$+ \rho H_{N_0, 1} - \rho H_{N_0-1, 1}$$

$$= \sum_{j=1}^{N_0} \rho H_{j, N_0+1-j} - \rho H_{j-1, N_0-j+1} \quad (6.4.3)$$

where

$$\rho \triangleq \text{rank of }$$

It should be noted that further work is necessary to show the existence of dimension $n(N_0)$ in all cases [93].

Solutions to the problem of minimal partial realization were
proposed by Tether [93] and Kalman [48] independently. Essentially the algorithms are based on the extension of the given sequence of Markov parameters such that Ho's algorithm [43] may be used to find a minimal realization. However it has been found that the extension to a given sequence may not be unique (this, in fact, is true in most cases) as the example given by Tether illustrates. Rossen [81, 82], has applied Tether's algorithm to the identification of complex chemical processes, with varying degrees of non-linearity.

Rossen [82] has suggested that the non-unique elements of the extension, of the given sequence, be chosen such that the system realization is stable. Unfortunately this is not a very satisfactory solution, since if there are three or more elements to be chosen, then it will be difficult to test for the stability of the realization in terms of the unknown coefficients.

The main difference between the partial realizability problem and the reduction problem is the fact that in the latter, an infinite sequence is available (since the given high order system may be expanded into an infinite sequence), but we require the model to fit at least the first $N_o$ Markov parameters (or time-moments) of the system. Thus, for the reduction problem, if a non-unique element is to be chosen, then we can choose it such that more of the original system's Markov parameters (time-moments) are fitted. In this way, we remove the problem of non-uniqueness, and at the same time make the reduced model approximate the system even closer. We have found that this form of minimal realization, for a given sequence of parameters of a given high order system, is most suitably accomplished by applying any of the algorithms [1, 43, 77, 89]. However for illustrative purposes, in Section 6.5, Silverman's algorithm will be described and then together with the theory of Section 6.2 applied to the reduction of high order systems.
6.5 Application of Minimal Realization Algorithms to the Reduction of Linear Systems

There are many algorithms for the minimal realization of systems based on the Hankel matrix of an infinite sequence [1, 43, 77, 89], and for the purpose of model reduction any one of them may be used. However, Silverman's algorithm has been found to be most suitable for illustrative purposes, since the parameters of the given sequence are explicitly retained in the realization, and thus it is easy to see how reduction is effected. The algorithm will be briefly outlined and a fuller description may be found in reference [89].

Define the sequence of Hankel matrices of the system as:

\[
H_{ij} = \begin{bmatrix}
D_1 & D_2 & \ldots & D_j \\
D_2 & D_3 & \ldots & D_{j+1} \\
\vdots & \vdots & \ddots & \vdots \\
D_i & D_{i+1} & \ldots & D_{i+j-1}
\end{bmatrix}
\] (6.5.1)

where the \(D_i\) (\(i = 1, 2, \ldots\)) are the given (pxq) sequence of Markov parameters of the system.

If the sequence of Markov parameters is generated by a triple \((A,B,C)\)_n with controllability and observability matrices \(Q_j\) and \(R_i\), then

\[
H_{ij} = R_i Q_j
\] (6.5.2)

This relationship underlies the role of the Hankel matrices in the realization problem.

Let \(Y_{ij}\) denote the \((i,j)\) element of the Hankel matrix. It is clear from the form of this matrix that

\[
Y_{i+p,j} = Y_{i,j+q} \quad \forall \ i, j, = 1, 2, 3, \ldots
\] (6.5.3)

where \(p\) and \(q\) are the number of outputs and inputs respectively.

Noting that the realization theorem (Theorem 6.1) has the condition that

"an infinite sequence is realizable if and only if there exists
non-negative integers $\beta, \alpha$, and $n$ such that for $j = 1, 2, ...$

$$\text{rank } H_{\beta, \alpha} = \text{rank } H_{\beta+1, \alpha+j} = n \quad (6.5.4)$$

Hence

$$\text{rank } H_{\beta+i, \alpha+j} = n \quad \forall \ i, j = 1, 2, 3, ... \quad (6.5.5)$$

Since the $(\beta+i)^{th}$ block of rows in $H_{\beta+i, \alpha+j}$ is contained in the $(\beta+i-1)^{th}$ block of rows in $H_{\beta+1, \alpha+j}$, by Equation (6.5.3).

Let $G_\alpha$ denote the submatrix formed from the first $n$ independent rows of $H_{\beta, \alpha}$, and let $G^\alpha_\alpha$ be the submatrix of $H_{\beta+1, \alpha}$ positioned $p$ rows below $G_\alpha$ (i.e. if the $i^{th}$ row of $G_\alpha$ is the $j^{th}$ row of $H_{\beta, \alpha}$ then the $i^{th}$ row of $G^\alpha_\alpha$ is the $(j+p)^{th}$ row of $H_{\beta+1, \alpha}$). The following four matrices are then uniquely defined as submatrices of $H_{\beta+1, \alpha}$.

$F_\alpha$: The non-singular $(nxn)$ matrix formed from the first $n$ independent columns of $G_\alpha$.

$F^\star_\alpha$: The $(nxn)$ matrix occupying the same column positions in $G^\alpha_\alpha$ as does $F$ in $G_\alpha$.

$F_1$: The $(pxn)$ matrix occupying the same column positions in $H_{1, \alpha}$ as does $F$ in $G_\alpha$.

$F_2$: The $(nxq)$ matrix occupying the first $q$ columns of $G_\alpha$.

Then it can be shown [89] that the triple

$$(F^\star_\alpha F^{-1}, F_2, F_1 F^{-1}) \quad (6.5.6)$$

defined from submatrices of $H_{\beta+1, \alpha}$ realizes the infinite sequence of Markov parameters, if Theorem 6.1 holds.

The application of the above algorithm to the problem of model reduction is best illustrated by applying it to the reduction of single input-single output systems. Thus let the high order system, of order $n$ say, be given in terms of its Markov parameters as follows

$$G(s) = \sum_{i=1}^{\infty} D_i s^{-i} \quad (6.5.7)$$

where the $D_i$ ($i = 1, 2, ...$) are scalar constants. Then defining the sequence of Hankel matrices as shown in Equation (6.5.1), and
using Theorem 6.1, Silverman's algorithm reduces to

\[ F = \begin{bmatrix} D_2 & D_3 & \ldots & D_{n+1} \\ D_3 & D_4 & \ldots & D_{n+2} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n+1} & D_{n+2} & \ldots & D_{2n} \end{bmatrix} \]  \hspace{1cm} (6.5.8)

\[ F^* = \begin{bmatrix} D_2 & D_3 & \ldots & D_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n+1} & D_{n+2} & \ldots & D_{2n} \end{bmatrix} \]  \hspace{1cm} (6.5.9)

\[ F_1 = F_2^T = [D_1 \ D_2 \ \ldots \ D_n] \]  \hspace{1cm} (6.5.10)

and the minimal realization \((A,B,C)_n\) is then given by Equation (6.5.6).

### 6.5.1 State-vector representation of reduced order models derived by Padé approximation about \(s = 0\)

Given the high order system, whether described by transfer function or in state-vector form, it can be expanded into a power series of the form

\[ G(s) = -\sum_{i=1}^{\infty} C_i s^{i-1} \]  \hspace{1cm} (6.5.11)

where the \(C_i\) (\(i = 1, 2, 3, \ldots\)) are the time-moments of the system, and are scalar constants.

Using Silverman's algorithm and the theory developed in Section 6.2.2, a minimal realization of (6.5.11) is given by the triple \((A,B,C)_n\) where

\[ A = (F^*F^{-1})^{-1} = F_1F^{-1} \]

\[ B = F_2 \]  \hspace{1cm} (6.5.12)

\[ C = (F_1^{-1}F_1F_2^{-1})F_2^{-1} = F_1F^{-1} \]

and the matrices \(F, F^*, F_1\) and \(F_2\) are as defined in (6.5.8) - (6.5.10), but with the \(D_i\)'s being replaced by the \(C_i\)'s of Equation (6.5.11).

Thus
A reduced model, of order k, is then given by the triple

\((A_k, B_k, C_k)\) where

\[ A_k = F_k \left( F^* \right)^{-1} \]
\[ B_k = (F_2)^k \]
\[ C_k = (F_1)^k \left( F^* \right)^{-1} \]

and the matrices \(F_k, F^*_k, (F_1)^k, \) and \((F_2)^k\) are obtained by partitioning \(F, F^*, F_1,\) and \(F_2\) as shown above.

It is easy to see that (6.5.13) has the same first 2k time-moments as the system. Further the reduced order model is equivalent to the \([k-1,k]\) Padé approximant of the system, and is therefore equivalent to the model derived by Chen and Sheih [19]. It should be noted that the matrices \(F, F^*, F_1\) and \(F_2\) are partitioned and not the triple \((A,B,C)\) of Equation (6.5.12). This form of reduction is useful computationally, since the algorithm only requires the inversion
of a \((k \times k)\) constant symmetric matrix. Higher order models may be derived by the addition of rows and columns. However, one constraint on the use of this method of reduction is that the reduced model is a \([k-1,k]\) type approximant.

6.5.2 State-vector representation of 'biased' reduced order models

In Chapter 3, reduced order models were derived such that the coefficients of their power series expansion about \(s = 0\) (\(s = \infty\)), agree with that of the system up to, and including, the term in \(s^{k+m-1}\) \((s^{-k+m})\) for some integer \(m\); \(k\) is the order of the reduced model. For notational simplicity we will assume \(m = 0\), i.e. the model is equally biased.

Let the high order system be given by

\[
G(s) = -\sum_{i=1}^{\infty} C_i s^{-i} \tag{6.5.14}
\]

\[
G(s) = \sum_{i=1}^{\infty} D_i s^{-i} \tag{6.5.15}
\]

Assuming \(G(s)\) is of order \(n\), then using Silverman's algorithm and the theory of Section 6.2.3, a minimal realization of the system, with \(m = 0\), is given by the triple \((A, B, C)\), where the matrices \(A, B,\) and \(C\) are as defined by Equation (6.5.12), and the matrices \(F, F^*, F_1\) and \(F_2\) are given by

\[
F = \begin{bmatrix}
D_n & D_{n-1} & \cdots & D_k & D_{k-1} & \cdots & D_2 & D_1 \\
D_{n-1} & D_{n-2} & \cdots & D_{k-1} & D_{k-2} & \cdots & D_2 & D_1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
D_{n-k+1} & D_{n-k+2} & \cdots & D_1 & D_{k-2} & \cdots & D_2 & D_1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
D_2 & D_1 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
D_1 & C_1 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\

\end{bmatrix}
\]
Again by partitioning $F, F^*, F_1,$ and $F_2$ as shown above, a reduced model, of order $k$, is obtained which 'fits' the first $k$ Markov parameters and the first $k$ time-moments of the given high order system. Using the theory developed in Section 6.2.3, reduced order models for the case $m \neq 0$, may also be obtained. It should be noted that the transfer function of the reduced order models are constrained to be of the $[k-1,k]$ type approximant.

We have only shown how Silverman's algorithm may be applied for the reduction of single input-single output systems, but in fact any one of the algorithms of minimal realization, which is based on the Hankel matrix approach, may be used.

6.5.3 Reduction of multivariable systems

The approach outlined in Sections 6.5.1 and 6.5.2 may be generalised to the case of multivariable systems, by deriving the matrices $F, F^*, F_1$ and $F_2$ for the system and then reduction is effected by suitably partitioning these matrices. The reduced models obtained using this approach do fit the initial time-moments (or Markov parameters or a combination of both) but may not necessarily be equivalent to the Padé approximants derived in Chapter 4. In this
section we will illustrate the method by considering two examples. The first one illustrates how the method may produce the same model as the Padé approximants of Chapter 4. The second example illustrates why the method may lead to different models to those of Chapter 4, because unlike the method of Chapter 4, the time-moments of the system may not all be fitted exactly.

**Example 1**

Let

\[
G(s) = \frac{(s+20)}{(s+1)(s+10)} \cdot \frac{(s+10)}{(s+2)(s+5)}
\]

The power series expansion of \( G(s) \) about \( s = 0 \), is given by

\[
G(s) = \frac{2}{1} \cdot \frac{2.1}{0.6} \cdot \frac{2.11}{0.32} \cdot \frac{2.111}{0.164} + \ldots
\]

Then form the Hankel matrix

\[
H_{ij} = \begin{bmatrix}
2 & 2.1 & 2.11 & -2.111 & \ldots \\
1 & -0.6 & 0.32 & -0.164 & \ldots \\
2.1 & 2.11 & -2.111 & \ldots & \ldots \\
-0.6 & 0.32 & -0.164 & \ldots & \ldots \\
2.11 & -2.111 & \ldots & \ldots & \ldots \\
0.32 & -0.164 & \ldots & \ldots & \ldots \\
\vdots & \vdots & \vdots & \vdots & \vdots 
\end{bmatrix}
\]

Suppose that \( R(s) \) is to be of order two. Hence take \( G_\alpha \) to be the first two linearly independent rows of \( H_{ij} \)

i.e.

\[
G_\alpha = \begin{bmatrix}
2 & 2.1 & 2.11 & -2.111 & \ldots \\
1 & -0.6 & 0.32 & -0.164 & \ldots 
\end{bmatrix}
\]

therefore

\[
G_\alpha^* = \begin{bmatrix}
-2.1 & 2.11 & -2.111 & \ldots \\
-0.6 & 0.32 & -0.164 & \ldots 
\end{bmatrix}
\]

Then \( F \) is the (2x2) matrix formed from the first two linearly independent columns of \( G_\alpha \).
Therefore
\[
F = \begin{bmatrix} 2 & -2.1 \\ 1 & -0.6 \end{bmatrix}
\]
\[
F^* = \begin{bmatrix} -2.1 & 2.11 \\ -0.6 & 0.32 \end{bmatrix}
\]
\[
F_1 = \begin{bmatrix} 2 & -2.1 \\ 1 & -0.6 \end{bmatrix}
\]
\[
F_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}
\]

Hence using the theory of Section 6.2.2, the state-vector representation of the model is given by
\[
\dot{x} = [F(F^*)^{-1}]x + F_2u
\]
\[
y = -F_1(F^*)^{-1}x
\]

substituting for $F,F^*,F_1$ and $F_2$ gives
\[
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1.04377 & +0.31987 \\ -0.06734 & -1.43098 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u
\]
\[
\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} +1.04377 & -0.31987 \\ +0.06734 & +1.43098 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
\]

The transfer function matrix $R(s)$, of the model is given by
\[
R(s) = C(sI-A)^{-1}B
\]
\[
= \frac{1.76767s+3.030309}{s^2+2.47475s+1.515154}
\]

which is exactly the same as that derived in Example 1, Section 4.7, Chapter 4, by the Padé approximation method. It is easy to show that the first three time-moments of $R(s)$ are equal to the first three time-moments of $G(s)$.

**Example 2**

Consider the following $(2 \times 2)$ system
\[
G(s) = \begin{bmatrix} \frac{s+4}{s^2+3s+2} & \frac{5}{s^2+6s+5} \\ \frac{S+5}{s^2+11s+10} & \frac{s+6}{s^2+5s+6} \end{bmatrix} = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}
\]
G(s) may be expanded into a power series, about s = 0, of the form

\[ G(s) = -C_1 - C_2 s - C_3 s^2 - C_4 s^3 - \ldots \]

where

\[
\begin{align*}
C_1 &= \begin{bmatrix} -2 & -1 \\ -0.5 & -1 \end{bmatrix}; & C_2 &= \begin{bmatrix} 2.5 & 1.2 \\ 0.45 & 0.66667 \end{bmatrix}; \\
C_3 &= \begin{bmatrix} -2.75 & -1.24 \\ -0.445 & -0.388889 \end{bmatrix}; & C_4 &= \begin{bmatrix} 2.875 & 1.248 \\ 0.4445 & 0.212962963 \end{bmatrix}; etc.
\end{align*}
\]

Then form the Hankel matrix

\[
H_{ij} = \begin{bmatrix}
-2 & -1 & 2.5 & 1.2 & -2.75 & -1.24 & 2.875 & 1.248 \\
-0.5 & -1 & .45 & 0.66667 & -0.445 & -0.388889 & 0.4445 & 0.21296 \\
2.5 & 1.2 & -2.75 & -1.24 & 2.875 & 1.248 & & \\
.45 & 0.66667 & -0.445 & -0.388889 & 0.4445 & 0.212963 & & \\
-2.75 & -1.24 & 2.875 & 1.248 & & & & \\
-0.445 & -0.388889 & 0.4445 & 0.212963 & & & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}
\]

Suppose we are interested in a third order model. Then \( G_\alpha \) is taken to be the first three linearly independent rows of \( H_{ij} \).

i.e.

\[
G_\alpha = \begin{bmatrix}
-2 & -1 & 2.5 & 1.2 & -2.75 & -1.24 & \ldots \\
-0.5 & -1 & 0.45 & 0.66667 & -0.445 & -0.388889 & \ldots \\
2.5 & 1.2 & -2.75 & -1.24 & 2.875 & 1.248 & \ldots \\
2.5 & 1.2 & -2.75 & -1.24 & & & \ldots \\
.45 & 0.66667 & -0.445 & -0.388889 & & & \ldots \\
-2.75 & -1.24 & 2.875 & 1.248 & & & \ldots \\
\end{bmatrix}
\]

\[
G_\alpha^* = \begin{bmatrix}
-2 & -1 & 2.5 \\
-0.5 & -1 & 0.45 \\
2.5 & 1.2 & -2.75 \\
2.5 & 1.2 & -2.75 \\
.45 & 0.66667 & -0.445 \\
-2.75 & -1.24 & 2.875 \\
\end{bmatrix}
\]

Then it follows from Silverman's algorithm that

\[
F = \begin{bmatrix}
-2 & -1 & 2.5 \\
-0.5 & -1 & 0.45 \\
2.5 & 1.2 & -2.75 \\
\end{bmatrix}
\]

\[
F^* = \begin{bmatrix}
-2.75 & -1.24 & 2.875 \\
-0.445 & -0.388889 & 0.21296 \\
\end{bmatrix}
\]
\[ F_1 = \begin{bmatrix} -2 & -1 & 2.5 \\ -0.5 & -1 & 0.45 \end{bmatrix}, \quad F_2 = \begin{bmatrix} -2 & -1 \\ -0.5 & -1 \\ 2.5 & 1.2 \end{bmatrix} \]

Hence using Equations (6.5.12), the reduced model is given by

\[
\begin{align*}
\dot{X} &= [F(F^*)^{-1}]x + F_2u \\
Y &= [F_1(F^*)^{-1}]x
\end{align*}
\]
or

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{x}_3 \\
\end{bmatrix} = 
\begin{bmatrix}
-2.95307 & 0.25145 & -1.91620 \\
0.21527 & -1.70391 & 0.09870 \\
1.0 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
\end{bmatrix} + 
\begin{bmatrix}
-2 & -1 \\
-0.5 & -1 \\
2.5 & 1.2 \\
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
\end{bmatrix}
\]

The matrix transfer function of the model, \( R(s) \), is given by

\[
R(s) = \frac{(0.98992s^2 + 5.68724s + 6.48044)(0.40218s^2 + 3.0055s + 3.24022)}{s^3 + 4.656984s^2 + 6.893844s + 3.24022}
\]

\( R(s) \) may be expanded into a power series, about \( s = 0 \), to give

\[
R(s) = \frac{2}{0.5} + \frac{2.75 \cdot 1.24}{0.45 \cdot 0.66667} s + \frac{2.875 \cdot 1.222}{0.454 \cdot 0.366} s^2 + \ldots
\]

Thus it can be seen that \( R(s) \) fits the first four time-moments of \( g_{11}(s) \), the first three time-moments of \( g_{12}(s) \) and \( g_{21}(s) \), and the first two time-moments of \( g_{22}(s) \).

Hence \( R(s) \) is biased in the sense that it approximates certain entries of \( G(s) \) more closely than others.

The step responses for individual entries of \( R(s) \) and \( G(s) \) were plotted. It was found that the step responses of \( g_{11}(s) \) and \( r_{11}(s) \) were very close together. The step responses of \( g_{12}(s) \) and \( r_{12}(s) \) were as shown in graph 6.1, while the step responses of \( g_{22}(s) \) and \( r_{22}(s) \) were as shown in graph 6.2. From the graphs it is easy to see that \( R(s) \)
approximates $g_{12}(s)$ more closely than $g_{22}(s)$, which is predictable since more of the time-moments of $g_{12}(s)$ are fitted.

If for this particular example, the reduced model $R(s)$ was required to fit the first three time-moments of $G(s)$ exactly, then using Lemma 6.1, it can be shown that the minimum order of $R(s)$ must be four. Hence a fourth order model, which fits exactly the first three time-moments of $G(s)$, may be found by a similar procedure to that above.
Chapter 7
LINEAR SYSTEM REDUCTION USING PADS APPROXIMATION
TO ALLOW RETENTION OF DOMINANT MODES

7.1 Introduction

In earlier Chapters we have developed the continued fraction and Padé-type techniques for the reduction in order of linear systems. It has been noted that a serious shortcoming of these methods is that the reduced model may be unstable even though the high order system is stable, and vice-versa.

A large number of methods of reduction [7, 24, 28, 60, 62, 68] are based on the retention of the dominant poles of the original system in the reduced model. The most important feature of these methods is that the reduced model is always stable if the original high order system is stable. However in almost all these methods, the system is considered in state-vector form and thus involve the computation of the eigenvalues and eigenvectors of the high order state matrix. This, computationally is very cumbersome and, in fact, it is known to fail when the eigenvalues of the system are widely separated [7].

Our purpose in this Chapter is to develop a method of reduction which combines the desirable features of the Padé approximation methods, Chapters 3, 4 and 6, and the modal methods of Section 2.3.1. The method is based on the concept of Padé approximation about more than one point (see Appendix II). The reduced order model is constrained to retain the dominant poles (or any other desirable poles) of the system. The rest of the coefficients of the reduced model are chosen to ensure that the reduced model has the same initial time-moments (or Markov parameters) of the system. The algorithm is
extended to the reduction of multivariable systems.

Thus given a transfer function of a system, the problem is finding its dominant poles (poles nearest the origin) or the largest magnitude poles. Nagarajan [66] suggests that the method of Eisenberg [32] should be used to determine the dominant poles of the system transfer function. In Section 7.2 an alternative method of computing the dominant poles of the system based on Koenig's theorem and its generalisation for the convergence of poles of Padé approximants [45], which is numerically more efficient, is introduced. The method has the very important added advantage of approximating a number of dominant modes in one operation, which is very useful in the problem of reducing the order of linear systems.

In Section 7.3, a new method of reduction is introduced which retains the dominant poles, or any desired poles, of the high order system in the reduced order model, using Koenig's theorem and its generalisation. The transfer function of the reduced order model is then chosen to approximate the full system in the Padé sense. The method may be applied to systems described by a transfer function or a state-vector representation. In Section 7.4 the method is extended to the problem of reducing the order of multivariable systems.

In Section 7.5 the method is applied to derive biased models, in the sense that the reduced order model may retain a number of large magnitude poles as well as the dominant poles of the system, hence better approximation is obtained at the initial transient response.

In Section 7.6 the advantages and disadvantages of the method are stated.

In Section 7.7, the problem of instability of the reduced model, encountered by the methods of Chapters 3 and 4 is discussed and an
algorithm based on the retention of one or more poles of the system in the reduced model, is introduced. The algorithm will always lead to a stable model.

7.2 Evaluation of Polynomial Zeros using Koenig's Theorem

The analysis of linear control systems depends to a great extent upon a knowledge of the manner in which the roots of the characteristic equation of the system vary with respect to its coefficients. An analysis of these root variations is complicated by the fact that for higher order systems there exists no closed form solutions for these roots. However, since the response of these systems is often dominated by one or two roots of small magnitude, an approximate solution for these roots would be advantageous. In this Section we will use Koenig's theorem, and its generalisation, for the convergence of poles of Padé approximants, to compute the approximate solution for the dominant roots of a given polynomial.

Koenig's theorem, and its generalisation, will be stated here, and the proofs may be found in references [41, 45].

Theorem 7.1 (Koenig)

Let

\[ f(s) = c_0 + c_1 s + c_2 s^2 + \ldots, \quad c_i \text{ real and } c_0 \neq 0 \quad (7.2.1) \]

be meromorphic for \(|s| < R\), and in this disc let it have just one simple pole at \(s = r\). If

\[ |r| < \sigma R < R, \]

then

\[ \frac{c_U}{c_{U+1}} = r + O(\sigma^{U+1}) \quad (7.2.2) \]
Theorem 7.2 (Generalisation)

Let \( f(s) \), given in (7.2.1), be meromorphic for \(|s| < R\), and let it have exactly \( p \) poles \( r_1, r_2, \ldots, r_p \), not necessarily distinct in this disc. Let

\[
0 < |r_1| \leq |r_2| \leq |r_3| \leq \ldots \leq |r_p| < \sigma R < R
\]

and let

\[
\Psi(s) = (1-r_1^{-1}s)(1-r_2^{-1}s) \cdots (1-r_p^{-1}s)
= 1 + a_1s + a_2s^2 + \ldots + a_ps^p
\]  

(7.2.3)

Finally, let the denominator of the \([U,p]\) Padé approximant be

\[
K_U(s) = 1 + a^{(U)}_1s + a^{(U)}_2s^2 + \ldots + a^{(U)}_ps^p
\]  

(7.2.4)

Then

\[
a^{(U)}_i = a_i + O(\sigma^U)
\]  

(7.2.5)

\[
K_U(s) = \Psi(s) + O(\sigma^U)
\]  

(7.2.6)

For the case when \( p = 1 \), Theorem 7.2 reduces to Theorem 7.1.

Given the polynomial

\[
F_n(s) = 1 + a_1s + a_2s^2 + \ldots + a_ns^n
\]  

(7.2.7)

to find its smallest magnitude root (roots), the following procedure is applied

I. The function \([F_n(s)]^{-1}\) is expanded into a power series of the form

\[
[F_n(s)]^{-1} = c_0 + c_1s + c_2s^2 + \ldots
\]  

(7.2.8)

where

\[
c_0 = 1
\]

\[
c_1 = -a_1c_0
\]

and in general

\[
c_i = - \sum_{j=1}^{i} a_jc_{i-j}, \quad a_j = 0 \quad \forall \quad j > n
\]  

(7.2.9)

II. Compute

\[
\delta_i = \frac{c_i}{c_{i+1}} \quad (i = 0, 1, 2, \ldots)
\]  

(7.2.10)

Using Theorem 7.1, it is easy to see that if there exists
a smallest magnitude real root, then \( \delta_1 \) will converge to it. If a pair of complex conjugate roots are dominant, then \( \delta_1 \) will not converge. If the smallest magnitude root is a multiple root, then \( \delta_1 \) will converge very slowly, and in that case step III should be carried out.

### III

If the smallest magnitude roots are complex conjugate, or multiple roots, then Koenig's generalised theorem may be used to obtain the \( p \) roots of the denominator of the \([u,p]\) Padé approximant, which will converge to the \( p \) smallest magnitude zeros of \( F_n(s) \) as \( u \to \infty \).

The coefficients of the denominator of the \([u,p]\) Padé approximant may be found as follows. Let the denominator be given by

\[
s^p + b_1 s^{p-1} + b_2 s^{p-2} + \ldots + b_p
\]  

(7.2.11)

The \( b_i \) (\( i = 1, 2, \ldots, p \)) may be computed by using the following relation (see Appendix II)

\[
\begin{bmatrix}
c_{u-p+2} & c_{u-p+3} & \cdots & c_{u+1} \\
c_{u-p+3} & c_{u-p+4} & \cdots & c_{u+2} \\
\vdots & \vdots & \ddots & \vdots \\
c_{u+1} & c_{u+2} & \cdots & c_{u+p} \\
\end{bmatrix}
\begin{bmatrix}
b_1 \\
b_2 \\
\vdots \\
b_p \\
\end{bmatrix} = 
\begin{bmatrix}
c_{u-p+1} \\
c_{u-p+2} \\
\vdots \\
c_{u} \\
\end{bmatrix}
\]  

(7.2.12)

Thus for the case when the smallest magnitude roots are a complex conjugate pair, then the denominator of the \([u,2]\) Padé approximant is given by

\[
K_{2,0}(s) = s^2 + b_{1,2} s + b_{2,2}
\]  

(7.2.13)

where

\[
b_{1,0} = \frac{c_{u+2} c_{u-1} - c_{u} c_{u+1}}{c_{u+1} c_{u} c_{u+2}}
\]  

(7.2.14)

\[
b_{2,0} = \frac{c_{u} c_{u+1} - c_{u-1} c_{u+2}}{c_{u+1} c_{u} c_{u+2}}
\]  

(7.2.15)

\( c_{-1} = 0 \)
The zeros of (7.2.7) will then converge to the least magnitude complex conjugate pair as \( \nu \to \infty \).

It is very easy to show that for an \( n \)th order polynomial, the approximate least magnitude root, \( r \), obtained by Eisenberg's method [32] is given by

\[
    r = \frac{c_{n-1}}{c_n}
\]

where \( c_{n-1} \) and \( c_n \) are as defined in (7.2.9). It can further be shown that calculating \( r \) using the above procedure would lead to a better approximation in comparison with Eisenberg's method. In fact the approximation may be made as accurate as required. The procedure has the added advantage of giving better approximations when the roots are close together, multiple, or complex conjugate. Further, it should be noted that the convergence of the above algorithm may be speeded up if the location of the dominant root is known approximately. Thus if the approximate value of the root is \( r_1 \), then a more accurate approximation (with much faster convergence) may be obtained by expanding \( [P_n(s+r_1)]^{-1} \) into a power series expansion and the above procedure applied.

Example 2 was chosen to illustrate this point. Further, the method outlined by Zakian [104], for the approximate solution of the least magnitude zero of a given polynomial, can be shown to be an application of Theorem 7.1.

**Example 1**

One of the examples discussed by Eisenberg was

\[
    F(s) = (s+1)(s^2 + 6s + 25)(s^2 + 10s + 61)
\]

\[
= s^5 + 17s^4 + 162s^3 + 762s^2 + 2141s + 1525
\]

Using the procedure outlined above, the following results are obtained.
\[ \delta_0 = -0.7122840 \\
\delta_1 = -0.9541748 \\
\delta_2 = -1.0006425 \\
\delta_3 = -1.00022180 \\
\delta_4 = -1.0005366 \\
\delta_5 = -1.0000398 \]

where \( \delta_4 \) is the approximate value obtained by Eisenberg. It can be seen that more accurate values may be obtained by using the above method.

**Example 2**

This example was chosen to illustrate how Theorem 7.2 may be applied and how the convergence of the algorithm speeded up.

\[ F(s) = (s+1)(s+2)(s+10)(s+20) = s^4 + 33s^3 + 292s^2 + 660s + 400 \]

Using the above procedure to find the smallest magnitude root, we get

\[ \delta_0 = -0.606061 \\
\delta_1 = -0.828105 \\
\delta_2 = -0.920058 \\
\delta_3 = -0.961482 \\
\delta_4 = -0.981097 \\
\delta_5 = -0.990636 \\
\delta_6 = -0.995340 \]

Hence Eisenberg's method does not produce an accurate approximation \( \{\delta_3\} \), while this procedure may produce as accurate an approximation as is required. In this case, however, since convergence is not very fast we can apply the following procedure to speed up convergence.

From above, let the root \( r, \) be approximated by

\[ r_1 = -0.92 \]

Then

\[ F(s+r_1) = s^4 + 29.32s^3 + 205.998s^2 + 203.39885s + 14.96849 \]

Using the above procedure, and replacing \( F(s) \) by \( F(s+r_1) \) we get
\[
\begin{align*}
\delta' &= -0.0735918 & \delta &= -0.993592 \\
\delta_1' &= -0.0795185 & \delta_1 &= -0.9995185 \\
\delta_2' &= -0.07996242 & \delta_2 &= -0.99996242 \\
\end{align*}
\]

Thus the algorithm converges to the exact value of \( r \) very quickly.

Suppose that for this example we require the two smallest magnitude roots. Then using Theorem 7.2 and Equations (7.2.14) and (7.2.15), we get

<table>
<thead>
<tr>
<th>( u )</th>
<th>( b_{2,u} )</th>
<th>( b_{1,u} )</th>
<th>( r_1(\text{appx.}) )</th>
<th>( r_2(\text{appx.}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.839834</td>
<td>2.827799</td>
<td>-1.014803</td>
<td>-1.812996</td>
</tr>
<tr>
<td>2</td>
<td>1.963630</td>
<td>2.962352</td>
<td>-1.001205</td>
<td>-1.961022</td>
</tr>
<tr>
<td>3</td>
<td>1.992235</td>
<td>2.992103</td>
<td>-1.000013</td>
<td>-1.991970</td>
</tr>
<tr>
<td>4</td>
<td>1.998395</td>
<td>2.998382</td>
<td>-1.000006</td>
<td>-1.998367</td>
</tr>
<tr>
<td>5</td>
<td>1.999674</td>
<td>2.999677</td>
<td>-1.000000</td>
<td>-1.999670</td>
</tr>
</tbody>
</table>

and it can be seen that the approximate roots \( r_1 \) and \( r_2 \) are converging to the roots of the polynomials very fast.

In conclusion, it has been shown how the smallest magnitude zero of a polynomial (the same algorithm may be applied to find the largest magnitude zero by expanding \([F_n(s)]^{-1}\) into a power series in terms of \( s^{-1} \)) may be evaluated using Koennig's theorem. It has been shown how a number of zeros may be computed together rather than individually. Further any zero of the given polynomial may be computed if an estimate of its magnitude is known.

7.3 Padé Approximation and Dominant Mode Reduction

Consider the following high order system transfer function
where \( s_i \) (\( i = 1, 2, \ldots, n \)) are the \( n \) poles of \( G(s) \). The transfer function \( G(s) \) can be expanded into a power series expansion about \( s = 0 \) of the form

\[
G(s) = c_0 + c_1 s + c_2 s^2 + \ldots
\]

where

\[
c_0 = \frac{d_0}{e_0},
\]

\[
c_1 = \frac{1}{c_0} (d_1 e_1 c_0)
\]

and in general

\[
c_k = \frac{1}{c_0} (d_k - \sum_{j=1}^{k} e_j c_{k-j}) , \quad \forall k > 0
\]

with \( d_k = 0 \) \( \forall k > n-1 \).

Assume that a reduced model of order \( k \), is required which retains the pole at \( s = -s_1 \), say. Let the reduced model \( R(s) \) of the form

\[
R(s) = \frac{a_0 + a_1 s + \ldots + a_{k-1} s^{k-1}}{b_0 + b_1 s + \ldots + b_{k-1} s^{k-1} + s^k}
\]

The orders of the numerators of \( R(s) \) and \( G(s) \) have been assumed to be one less than the denominator to simplify the notation.

Then for \( R(s) \) to be a Padé approximant of \( G(s) \), we have (see Appendix II)

\[
a_0 = b_0 c_0
\]

\[
a_1 = b_0 c_1 + b_1 c_0
\]

\[
\ldots \ldots \ldots \ldots
\]

\[
o = b_0 c_2k-2 + b_1 c_2k-3 + \ldots + c_{k-1}
\]

\[
o = b_0 c_{2k-1} + b_1 c_{2k-2} + \ldots + c_k
\]

But, since \( R(s) \) is to have a pole at \( s = -s_1 \), then using the
concept of Padé approximation about more than one point (see Appendix II), the last equation of (7.3.6) is replaced by the following equation

\[ 0 = b_0 - b_1 s_1 + b_2 s_1^2 - \ldots + (-1)^k s_1^k \]  

(7.3.7)

Hence these equations can be solved for the coefficients \( b_i, a_i \) (\( i = 0, 1, \ldots, k-1 \)) of Equation (7.3.5).

Now suppose that a reduced order model, of order \( k \) say, is required which retains the \( k \) dominant poles (the \( k \) poles nearest the origin) of the high order system.

Further, suppose that the \( k \) dominant poles are known. The transfer function of the reduced order model can then be written as

\[ R(s) = \frac{a_0 + a_1 s + \ldots + a_k s^k - 1}{(s + s_1)(s + s_2) \ldots (s + s_k)} \]

(7.3.8)

where the \( b_i \) (\( i = 0, 1, \ldots, k-1 \)) may be computed in terms of \( s_1, s_2, \ldots, s_k \).

Then if \( R(s) \) is to approximate \( G(s) \), in the Padé sense, about \( s = 0 \), then the \( a_i \) (\( i = 0, 1, 2, \ldots, k-1 \)) may be determined using the first \( k \) equations of (7.3.5).

So far it has been assumed that the dominant poles of the system are known, which in most cases is not necessarily true. This is where Koenig's theorem and its generalisation is of great use. Any transfer function can be expanded into a power series about \( s = 0 \). Hence, by using Koenig's theorem we can determine the number of dominant poles and their locations very easily (as was shown in Section 7.2). Thus for example suppose that a system transfer function has the pole locations shown in Figure 7.1.
Then applying Koenig's theorem to the expansion
\[ G(s) = c_0 + c_1 s + c_2 s^2 + \ldots \]
the first pole nearest the origin, \( s_1 \), can be picked out. If \( s_1 \) is very close to \( s_2 \) and \( s_3 \), then the ratio \( \delta_i = \frac{c_i}{c_{i+1}} \) (see Theorem 7.1) will converge very slowly, which will imply that there is at least another pole close to \( s_1 \). Thus Theorem 2.2 is applied to test for the convergence of two poles. In this example it will not converge since \( s_2 \) and \( s_3 \) are complex conjugate. Hence the theorem is applied to test for the convergence of the first three poles. If the convergence is fast, then it will imply that the system has three dominant poles, and the equations which they satisfy can be easily computed (using Equation 7.2.12).

The rate of convergence may be used to determine the ratio between the smallest neglected pole and the largest retained pole. The example in Section 7.3.2 was chosen to illustrate this point.

Thus to reduce a high order transfer function, it is first expanded into a power series, then Koenig's theorem and its generalisation are used to determine the number of dominant poles and their locations, and then Padé approximation is used to determine the numerator coefficients of the reduced order transfer function. The amount of computation involved in using this method is the same as that required for ordinary Padé approximation except...
perhaps more coefficients of the series (7.3.3) may have to be computed.

It should be noted that common poles and zeros in $G(s)$ are automatically cancelled when using this method of reduction, and have no effect on the reduced order model.

It has been assumed that the system is given in transfer function form, however it has been shown in Section 3.3.1, how a system described in a state-vector form may be expanded into a power series expansion. Hence the method outlined in this Chapter is equally applicable to systems described in a state-vector form.

In the case when the system being modelled is unstable, then it is important that the reduced model should also be unstable. Hence the unstable modes of the original system must be retained in the reduced model. Koenig's theorem and its generalisation may be used to compute the unstable modes as follows.

Given $G(s)$, the following transformation is carried out

$$s = \frac{z^{-1}}{z+1}$$

(7.3.9)

to get $G(z)$. The unstable poles of $G(s)$ are then mapped outside the unit circle in the $z$-plane. Expand $G(z)$ in the form

$$G(z) = d_0 + \frac{d_1}{z} + \frac{d_2}{z^2} + \frac{d_3}{z^3} + \ldots$$

(7.3.10)

Then applying Koenig's theorem and its generalisation to (7.3.10), we get all the large magnitude poles of $G(z)$, which in this case will be the poles outside the unit circle. Having computed the unstable poles, the coefficients of the reduced model are computed as before.

7.3.1 An alternative interpretation using partial fractions

A useful interpretation of this method of reduction, is to consider the system transfer function in partial fraction form.
Thus let

\[ G(s) = \frac{x_1}{1-\alpha_1 s} + \frac{x_2}{1-\alpha_2 s} + \ldots + \frac{x_n}{1-\alpha_n s} \]  

(7.3.11)

Then, reduction, in its simplest form, would be to keep the first \( k \) terms and to neglect the others. Rewriting Equation (7.3.11) gives

\[
G(s) = x_1 (1+\alpha_1 s^{2}+\ldots) + x_2 (1+\alpha_2 s^{2}+\ldots) \\
+ \ldots + x_n (1+\alpha_n s^{2}+\ldots)
\]

\[= c_0 + c_1 s + c_2 s^2 + \ldots\]

where

\[ c_i = \sum_{j=1}^{n} x_j \alpha_j^i, \quad i = 0, 1, 2, \ldots \]

Let the reduced model be of the form

\[ R(s) = \frac{y_1}{1-\alpha_1 s} + \frac{y_2}{1-\alpha_2 s} + \ldots + \frac{y_k}{1-\alpha_k s} \]  

(7.3.12)

where \( k \) is the order of the reduced model. Then

\[ R(s) = y_1 (1+\alpha_1 s^{2}+\ldots) + y_2 (1+\alpha_2 s^{2}+\ldots) \\
+ \ldots + y_k (1+\alpha_k s^{2}+\ldots)
\]

\[= c'_0 + c'_1 s + c'_2 s^2 + \ldots\]

where

\[ c'_i = \sum_{j=1}^{k} y_j \alpha_j^i, \quad i = 0, 1, 2, \ldots \]

Since \( R(s) \) is to approximate \( G(s) \) in the Padé sense, we require

\[ c'_i = c_i, \quad i = 0, 1, 2, \ldots, (k-1) \]  

(7.3.13)

which will lead to the following set of equations.
The above interpretation is a useful one, since it reveals that the effect of the neglected poles \(\alpha_{k+1}, \ldots, \alpha_n\) is taken into account when computing the residues \(y_i\) \((i = 1, 2, \ldots, k)\). It should further be noted that for asymptotically stable systems both the reduced order model and the high order system have the same time-moments, as far as, and including, the \((k-1)\)th time-moment. Further there is no steady state error between the reduced model and the system for the inputs \(a_i t^i\), \(i = 0, 1, 2, \ldots, k-1\).

### 7.3.2 Examples

The examples in this Section illustrate how the method may be used, and some of the results are compared with those obtained by other similar methods.

**Example 3**

Consider the transfer function

\[
G(s) = \frac{(s+1.5)(s+4)}{(s+1)(s+2)(s+6)}
\]
\[
\frac{s^2 + 5.5s + 6}{s^3 + 9s^2 + 20s + 12}
\]

G(s) is then expanded into a power series of the form

\[G(s) = 0.5(c_0 + c_1 s + c_2 s^2 + \ldots)\]

where

\[
\begin{align*}
  c_0 &= 1.000000 \\
  c_1 &= -0.750000 \\
  c_2 &= 0.566667 \\
  c_3 &= -0.631944 \\
  c_4 &= 0.615741 \\
  c_5 &= -0.607832 \\
  c_6 &= 0.603910
\end{align*}
\]

Let

\[
\delta_i = \frac{c_i}{c_{i+1}}
\]

Then from above it is easy to see that \(\delta_i > 1.0\), which is the value of the dominant pole. Now consider the following relation

\[
\beta_i = \frac{\delta_i - \delta_{i+1}}{\delta_{i+1} - \delta_{i+2}} \quad i = 0, 1, 2, \ldots
\]

which may be rewritten as

\[
\beta_i = \frac{c_i}{c_{i+1}} - \frac{c_{i+1}}{c_{i+2}} = \frac{c_{i+2}}{c_{i+1}} - \frac{c_{i+1}}{c_{i+3}} = \frac{c_{i+3}}{c_{i+1}} - \frac{c_{i+1}}{c_{i+2}} = \frac{1}{\delta_{i+2}} - \frac{1}{\delta_{i+1}}
\]
Let the two least magnitude poles of \( G(s) \) be denoted by \( r_1 \) and \( r_2 \). Then by Koenig's theorem we have

\[
\lim_{i \to \infty} r_i = r_1 = \lim_{i \to \infty} \delta_i
\]

Also using Koenig's generalised theorem 7.2, together with Equation (7.2.15) we have

\[
\lim_{i \to \infty} \frac{r_i r_{i+1}}{r_{i+2}} = \lim_{i \to \infty} \frac{c_{i+1} - c_i c_{i+2}}{c_{i+2} - c_{i+1} c_{i+3}}
\]

Hence it follows that

\[
\lim_{i \to \infty} \beta_i = \lim_{i \to \infty} \frac{\left( r_i r_{i+1} / r_{i+2} \right)}{\left( r_{i+1} r_{i+2} / r_{i+3} \right)} = \frac{r_1 r_2}{r_2}
\]

Therefore

\[
\lim_{i \to \infty} \beta_i = \frac{r_2}{r_1}
\]

For this example, it was found that \( \beta_1 = 2.0 \), hence it follows that there is a pole at \( s = -2 \).

Theorem 7.2 is then used to find the two most dominant poles of \( G(s) \). Let the denominator of the \([U,2]\) Padé approximant be

\[
K_U(s) = s^2 + b_{1,U} s + b_{2,U}
\]

Then \( b_{1,U} \) and \( b_{2,U} \) may be computed using Equations (7.2.15) and (7.2.16). Thus computing the denominator of the \([U,2]\) Padé approximant, \( U = 1, 2, \ldots \), we have

<table>
<thead>
<tr>
<th>( U )</th>
<th>( b_{1,U} )</th>
<th>( b_{2,U} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.470588</td>
<td>3.529412</td>
</tr>
<tr>
<td>2</td>
<td>3.636364</td>
<td>2.649351</td>
</tr>
<tr>
<td>3</td>
<td>3.234867</td>
<td>2.237288</td>
</tr>
<tr>
<td>\Rightarrow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>3.000113</td>
<td>2.000113</td>
</tr>
</tbody>
</table>
Now consider the following relation

$$\gamma_u = \frac{b_{2,u} - b_{2,u+1}}{b_{2,u+1} - b_{2,u+2}} \quad u = 1, 2, \ldots$$

It can be shown that $\gamma_u \approx \frac{x_3}{x_2}$ (by a similar analysis to that above).

For this example $\gamma_u \approx 3.0$, hence it follows that there is a pole at $s = -6$.

Thus taking the dominant roots to be at $s = -1.0$ and $s = -2.0$, the following reduced order model is obtained:

$$R(s) = \frac{a_0 + a_1 s}{2.0 + 3.0s + s^2}$$

where

$$a_0 = 0.5(b_0 c_0) = 1.0$$

$$a_1 = 0.5(b_0 c_1 + b_1 c_0) = 0.75$$

Hence

$$R(s) = \frac{1.0 + 0.75s}{2.0 + 3.0s + s^2}$$

The step responses of $R(s)$ and $G(s)$ were plotted as shown in Graph 7.1. From the graph it can be clearly seen that $R(s)$ is a good approximation to $G(s)$. It should be noted that the roots $s_1 = -1.0$ and $s_2 = -2.0$ do not have to be known exactly for a good approximation to be obtained.

Example 4

This example was chosen to compare the responses of the reduced order model obtained by this method with the responses of reduced models obtained by Davison and Chidambaram [24]. Consider the transfer function

$$G(s) = \frac{1}{1+s} + \frac{2}{2+s} - \frac{6}{3+s} + \frac{4}{4+s}$$

Using the method outlined in this Section and retaining the two dominant modes, we get

$$R(s) = \frac{1.166}{1+s} - \frac{0.332}{2+s}$$

This can be compared with Davison's model [D2] and Chidambaram's model [C2] [24].
\[ [R(s)]_{D2} = \frac{11/18}{1+s} + \frac{7/36}{2+s} \]

\[ [R(s)]_{C2} = \frac{5/6}{1+s} + \frac{1/12}{2+s} \]

The step responses of \( G(s) \), \( R(s) \), \( [R(s)]_{D2} \) and \( [R(s)]_{C2} \) were plotted as shown in Graph 7.2. \( R(s) \) can be seen to be a good approximation to \( G(s) \) and is better than either of the other two methods.

Example 5

An important requirement of a reduction method is that successively higher order approximants should be closer to the high order system. This example was chosen to illustrate that the method of this Section satisfies this constraint. Consider the transfer function

\[ G(s) = \frac{a(s)}{b(s)} \]

where

\[ a(s) = 18s^7 + 514s^6 + 5982s^5 + 36380s^4 + 122664s^3 + 222088s^2 + 185760s + 40320 \]

\[ b(s) = s^8 + 36s^7 + 546s^6 + 4536s^5 + 22449s^4 + 67284s^3 + 118124s^2 + 109584s + 40320 \]

Using the method of this Section, the following successive approximants were obtained

2nd order model

\[ R_2(s) = \frac{2+6.7786s}{(s+1)(s+2)} \]

3rd order model

\[ R_3(s) = \frac{5+22.3357s+11.4441s^2}{(s+1)(s+2)(s+3)} \]

4th order model

\[ R_4(s) = \frac{24+95.34295s+68.1123s^2+15.1794s^3}{(s+1)(s+2)(s+3)(s+4)} \]

The step responses of \( G(s) \), \( R_2(s) \), \( R_3(s) \) and \( R_4(s) \) were plotted as shown in Graph 7.3. From the graph it can be seen that the
approximation improves as the order of \( R(s) \) is increased.

7.4 Reduction of Multivariable Systems

In Chapter 4, the Padé approximation technique and the continued fraction methods were extended to the problem of reducing the order of multivariable systems. However the methods may involve large amounts of computation, which may make the reduction of the system less desirable. In this Section the method outlined in Section 7.3, for the reduction of single input-single output systems, is extended to the problem of reducing the order of a multivariable system transfer function.

For a multivariable system

\[
Y(s) = G(s)u(s) \quad (7.4.1)
\]

where \( Y \) is the output \( p \)-vector and \( u \) is the input \( q \)-vector and \( G(s) \) is the \((p \times q)\) matrix transfer function of the system. Equation (7.4.1) may be rewritten in the form

\[
Y(s) = \frac{B_0 + B_1 s + B_2 s^2 + \ldots + B_{n-1} s^{n-1}}{a_0 + a_1 s + a_2 s^2 + \ldots + a_n s^n} u(s) \quad (7.4.2)
\]

where \( B_i \) (\( i = 0, 1, \ldots, n-1 \)) are \((p \times q)\) constant matrices, and the \( a_i \) (\( i = 0, 1, \ldots, n \)) are constant scalars. \( G(s) \) can thus be expanded into a power series of the form

\[
G(s) = C_0 + C_1 s + C_2 s^2 + \ldots \quad (7.4.3)
\]

where the \( C_i \) (\( i = 0, 1, 2, \ldots \)) are \((p \times q)\) constant matrices which satisfy the relation

\[
C_i = \frac{1}{a_0} \left[ B_i + \sum_{j=0}^{i-1} a_{i-j} C_j \right], \quad i = 0, 1, 2, \ldots \quad (7.4.4)
\]

with \( C_{-1} = 0 \) and \( B_i = 0 \) \( \forall i \geq n \).

Thus using Equation (7.4.4), the matrix transfer function may be expanded into a power series.
Let the reduced order model have a matrix transfer function of the form

\[
R(s) = \frac{A_0 + A_1 s + \ldots + A_{k-1} s^{k-1}}{b_0 + b_1 s + b_2 s^2 + \ldots + b_{k-1} s^{k-1}}
\]  

(7.4.5)

where the \( A_i \) (i = 0, 1, ..., k-1) are (p\times q) constant matrices, and the \( b_i \) (i = 0, 1, ..., k-1) are constant scalars.

The method of Section 7.3 is then applied, and the dominant modes of \( G(s) \) are retained in \( R(s) \) and the numerator coefficients of \( R(s) \) are chosen such that \( R(s) \) approximates \( G(s) \) in the Padé sense.

The procedure is as follows:

I. The least common denominator, \( d(s) \), say, of \( G(s) \) is found;

II. Koenig's theorem and its generalisation is then applied to the power series expansion of \([1/d(s)]\) to find the k dominant poles of \( G(s) \). This then determines the coefficients \( b_i \) (i = 0, 1, 2, ..., k-1) in (7.4.5);

III. The numerator matrices \( (A_i) \) of \( R(s) \) are then computed as follows

\[
A_0 = b_0 C_0
\]

\[
A_1 = b_0 C_1 + b_1 C_0
\]

\[
\ldots \ldots \ldots \ldots
\]

\[
A_{k-1} = b_0 C_{k-1} + b_1 C_{k-1} + \ldots + b_{k-1} C_0
\]

Thus \( R(s) \) has been chosen such that its first k time-moments are equal to the first k time-moments of \( G(s) \).

**Example 6**

Consider the following system [22]

\[
G(s) = \frac{14.96(s+1.7)(s+100) + 95150(s+1.898)(s+10)}{85.20(s+1.44)(s+100) + 124000(s+2.077)(s+10)}
\]

\[
G(s) \quad \text{is expanded into a power series of the form}
\]

\[
G(s) = C_0 + C_1 s + \ldots
\]
where

\[
C_0 = \begin{bmatrix}
1.0072079 & 715.2265347 \\
4.8589307 & 1006.348510 \\
\end{bmatrix}
\]

\[
C_1 = \begin{bmatrix}
-0.7946790 & -543.8277404 \\
-3.3175966 & -796.5872320 \\
\end{bmatrix}
\]

Koenig's generalised theorem was applied to the power series expansion of \([1/d(s)]\), and after two iterations, \(J = 2\), the dominant roots were found to approximately satisfy

\[
s^2 + 3.1975464s + 2.4915755 = 0
\]

This was then taken to be the denominator of \(R(s)\). The numerator matrices were then computed using Equation (7.4.6)

\[
A_0 = b_0C_0 = \begin{bmatrix}
2.5095345 & 1782.049018 \\
12.1063927 & 2492.4438514 \\
\end{bmatrix}
\]

\[
A_1 = b_0C_1 + b_1C_0 = \begin{bmatrix}
1.2405903 & 931.982157 \\
7.2706140 & 1213.903562 \\
\end{bmatrix}
\]

Hence

\[
R(s) = \frac{(2.5095345+1.2405903s) (1782.049018+931.982157s)}{s^2+1.1975464s+2.4915755}
\]

The step responses of \(G(s)\) and \(R(s)\) were found to be almost the same, and some of the values are shown below

<table>
<thead>
<tr>
<th>time (secs.)</th>
<th>original plant ((y_1))</th>
<th>simplified ((y_1))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>524.153</td>
<td>523.466</td>
</tr>
<tr>
<td>2.0</td>
<td>665.681</td>
<td>665.040</td>
</tr>
<tr>
<td>3.0</td>
<td>702.948</td>
<td>702.743</td>
</tr>
<tr>
<td>4.0</td>
<td>712.745</td>
<td>712.694</td>
</tr>
</tbody>
</table>

7.5 Biased Reduced Order Models

So far, in this Chapter, it has been assumed that the reduced
order model retains the dominant modes of the system and fits the initial time-moments of the system. Both these assumptions ensure that the reduced order model approximates the steady state response of the system. Thus there are two ways of improving the approximation to the initial transient response of the system.

1. By making the reduced model fit a number of Markov parameters of the system. Hence the coefficients of the numerator of the reduced model are chosen such that $R(s)$ fits the initial time-moments and initial Markov parameters of the system.

2. By constraining the reduced model to retain a number of the large magnitude poles of the system.

The first approach has been discussed in detail with respect to the Padé approximation technique in Chapters 3 and 4, and hence will not be discussed here. The second approach will be discussed in this Section, and it will be shown how Koenig's theorem may be used to compute the smallest and largest magnitude poles of the system.

Atary [7] has pointed out that in the reduction of a high order system, consisting of subsystems which differ so much in their dynamics, the neglecting of the high modes can lead to the omission of complete systems. Thus it is sometimes necessary to retain the high order modes of the original system in the reduced model. In reference [55] a generalisation of Davison's method [28], is made so that the initial transient response of the high order system may be approximated as well as the steady state response. The method is not very practical since the system response is approximated by three reduced order models. The first approximates the initial transient response of the system (hence contains the high modes of the system), the second approximates the response of the system at
intermediate times (hence contains the intermediate modes of the system), and the third approximates the response of the system at large times (hence contains the small modes of the system).

We will modify the method of Section 7.3 such that the reduced model retains both small and large magnitude poles of the system. The case of single input-single output systems is considered, and it may be generalised to the multivariable case by a similar procedure to that of Section 7.4.

Let the high order system be given by

\[
G(s) = \frac{d_0 + d_1 s + d_2 s^2 + \ldots + d_{n-1} s^{n-1}}{e_0 + e_1 s + e_2 s^2 + \ldots + e_{n-1} s^{n-1} + s^n}
\]

which can be expanded into a power series expansions about \( s = 0 \) and \( s = \infty \) of the form

\[
G(s) = c_0 + c_1 s + c_2 s^2 + \ldots
\]

\[
G(s) = \frac{m_1}{s} + \frac{m_2}{s^2} + \frac{m_3}{s^3} + \ldots
\]

where the \( c_i \) (\( i = 0, 1, 2, \ldots \)) are proportional to the time-moments and the \( m_i \) (\( i = 1, 2, \ldots \)) are the Markov parameters of the system. For simplicity, suppose we require a second order model \( R(s) \) which retains the highest and the lowest modes of the system. Let

\[
R(s) = \frac{a_0 + a_1 s}{b_0 + b_1 s + s^2}
\]

Using Koenig's theorem, we have

\[
\lim_{i \to \infty} \frac{c_i}{c_{i+1}} = p_s \text{ A smallest magnitude pole}
\]

\[
\lim_{i \to \infty} \frac{m_i}{m_{i+1}} = p_l \text{ A largest magnitude pole}
\]

Further suppose that \( R(s) \) is required to fit the first time-moment and the first Markov parameter of the system. Thus

\[
a_0 = b_0 c_0
\]

\[
a_1 = m_1
\]
The denominator of $R(s)$ is then given by
\[ b_0 + b_1 s + s^2 = (s + p_s)(s + p_k) \]
\[ \Rightarrow b_0 = p_s p_k \]
\[ b_1 = p_s + p_k \]

Hence using Equations (7.5.5) and (7.5.6), $p_s$ and $p_k$ are determined and thus $b_0$, $b_1$ computed. Equations (7.5.7) are then used to compute the $a_i$ $(i = 0, 1)$.

It is easy to show how intermediate poles may also be retained in the reduced model, so that the reduced model response will approximate the system for small, intermediate and large time as was suggested in reference [55].

7.6 Advantages and Disadvantages of the Method

The purpose in introducing the method of this Chapter has been to derive a method of reduction which retains the advantages of the modal methods (Section 2.3.1) and the advantages of the Padé approximation type methods (Chapters 3, 4 and 6). The method of this Chapter has, in general, the following advantages:

1. It is very easy to use;
2. Computationally it is simpler than other similar methods which retain the dominant poles;
3. Account is taken of the neglected poles, unlike other methods which simply neglect far-off poles;
4. The reduced models have a "measure of goodness" since for asymptotically stable systems, the method equates the first $k$ time-moments of the reduced model and the system ($k$ is the number of coefficients of the reduced model), or equivalently the first $k$ error coefficients of the reduced model and the system. Hence the reduced model approximates
the original system for input polynomials in time;
5. For the example considered, the response of the \( k^{th} \) order model appears to be a better approximation than that of the \( (k-1)^{th} \) order model;
6. The method may be applied to derive biased reduced order models;
7. The reduced model is stable (unstable) if the system is stable (unstable);
8. The method may be applied to "stiff" systems.

Thus although the method retains the advantages of the modal methods and the Padé approximation methods, it has also overcome some of the disadvantages of both procedures.

In the above analysis we have retained the least magnitude poles in the reduced model. This would give a good approximation in most cases. However in the case of oscillatory systems, that is systems which have large magnitude poles with small negative real parts, the retention of the smallest magnitude poles may lead to bad approximations. In that case it may be necessary to retain the poles with the smallest negative real part. This may be achieved by using Koenig's theorem as follows.

Consider the following system of Figure 7.2a

\[
\begin{array}{c}
\text{Fig. 7.2a Location of Poles in } s\text{-Plane} \\
\text{Using Koenig's theorem we can compute } s_1, \text{ as in Section 7.2.}
\end{array}
\]
The following transformation is made

\[ p = s + s_1 \]  

(7.6.1)

The locations of the poles of the system, in the \( p \)-plane are as shown in Figure 7.2b.

Then the poles \( p_2 \) and \( p_3 \) (which correspond to \( s_2 \) and \( s_3 \)) lie in the right half plane. Hence the following bilinear transformation is made

\[ z = \frac{1+p}{1-p} \]  

(7.6.2)

which maps the left half plane (in the \( p \)-plane) into the unit circle in the \( z \)-plane as shown in Figure 7.2c.

From Figure 7.2c it can be seen that \( z_2 \) and \( z_3 \) (which correspond to \( s_2 \) and \( s_3 \)) are the largest magnitude poles of \( G(z) \). Hence \( G(z) \) is expanded into a power series expansion of the form

\[ G(s) = d_0 + \frac{d_1}{z} + \frac{d_2}{z^2} + \frac{d_3}{z^3} + \ldots \]  

(7.6.3)

Then using Koenig's generalised theorem on the Equation (7.6.3), the two largest magnitudes poles \( z_2 \) and \( z_3 \) may be computed.
the transformations (7.6.1) and (7.6.2) the poles $s_2$ and $s_3$ are computed.

The poles $s_2$ and $s_3$ are retained in the reduced model, and the rest of the coefficients of the reduced model are computed as before.

Bosley and Lees [14] have considered the problem of modelling the response of a high order oscillatory system. The method used is the fitting of time-moments of the system and reduced model. The reduced model is taken to be of the form

$$R(s) = \frac{e^{-Ts}}{b_0 + b_1 s + s^2}$$

the constants $b_0$, $b_1$, and $T$ are determined such that $R(s)$ fits the first three time-moments of the system. A disadvantage of the above type of model is that the exponential term must be approximated by a Padé approximant, hence raising the order of the model.

The following example was chosen to illustrate how the method of this Chapter will produce a stable model, when Padé approximation failed to do so.

**Example 7**

Consider the system

$$G(s) = \frac{136s^3 + 680s^2 + 952s + 408}{(s^2 + 2s + 102)(s + 2)^2}$$

It is easy to show that a second order model, derived by Padé approximation about $s = 0$ is unstable. The second order model derived by retaining the two complex conjugate poles and fitting the first two time-moments, is given by

$$R(s) = \frac{102 + 136s}{102 + 2s + s^2}.$$
The step responses of $G(s)$ and $R(s)$ are as shown in graph 7.4.

Chapman [18] has used the method of this Chapter to derive reduced order models, which were then used to speed up the convergence of the optimisation of a high order system.

In the case of multivariable systems the method of this Chapter has the following advantages over the methods of Chapter 4 and 6.

1. The method produces unique reduced order models;
2. The order of the denominator of the reduced model may be fixed prior to the reduction;
3. The reduced model is always stable if the high order system is stable;
4. Using Padé approximation, the reduced order model may be of high order [36]. For example, consider a system with $p = q = 3$, and let the reduced model fit the first four time-moments of the system. Then the degree of the characteristic equation of $R(s)$ may be as high as six. If the method of this Section is applied, and retaining the first four dominant modes of the system, the model can be made to fit the first four time-moments of the system.

Computationally, the method in general is very simple if the (l.c.d.) of $G(s)$ is known. If this has to be computed then the methods of Chapter 4 may become computationally easier.

A disadvantage of the method is that in the case of systems which are described by an irrational transfer function, a set of
dominant modes may not exist, and Koenig's theorem and its generalisation can not be applied.

7.7 The Derivation of Stable Reduced Order Models using Padé Approximation

It has been recognised that the Padé approximation technique, or equivalently the continued fraction methods, for the reduction of high order systems may produce unstable reduced order models, even though the high order system is stable [15, 27, 44, 82].

Brown [15] introduced an auxiliary performance criterion which stabilises the model, but this is done at the expense of increasing the order of the model. Rossen [82] introduced a procedure to overcome the instability of the reduced model. The procedure consists of determining the unstable model and isolating the positive pole. This pole is then replaced by a number of arbitrary poles and the system is constrained to fit the given Markov parameters (or time-moments) of the system. Then if the order of the model is too large it is suggested that Davison's method [28] should be used to retain the dominant poles of the model. Although this procedure produces a stable model, it is however computationally unattractive, since the poles of the unstable model have to be computed and the method in effect involves two reductions. It should be noted that the procedure is useful in identification where the number of available Markov parameters or time-moments is limited.

The following procedure is more suitable for the reduction problem, since we can generate as many Markov parameters or time-moments of the system as are required. The procedure can never fail to produce a stable model.

Given a high order system transfer function $G(s)$, as shown in
Equation (7.5.1), let it have the power series expansions (7.5.2) and (7.5.3). Further, let the reduced order model \( R(s) \) be of the form

\[
R(s) = \frac{a_0 + a_1 s + \ldots + a_{k-1} s^{k-1}}{b_0 + b_1 s + \ldots + b_{k-1} s^{k-1}} s
\]  

(7.7.1)

Then for \( R(s) \) to be a Padé approximant, about \( s = 0 \), of \( G(s) \), the \( 2k \) simultaneous equations (7.3.6) have to be satisfied. The coefficients \( a_i, b_i \) \( (i = 0, 1, \ldots, k-1) \) are then computed using Equation (7.3.6). If \( R(s) \) is found to be unstable, then it is constrained to retain one pole of the original system. Thus Koenig's theorem may be applied to the expansions (7.5.2) or (7.5.3) to compute the least magnitude or largest magnitude pole of \( G(s) \), and \( R(s) \) is then constrained to retain this pole. Thus using the concept of Padé approximation about more than one point, the last equation of the set of Equations (7.3.6) is replaced by the following equation

\[
0 = b_0 - b_1 s_1 + b_2 s_1^2 - \ldots + (-1)^k s_1^k
\]  

(7.7.2)

where \( s_1 \) is the pole to be retained in \( R(s) \). Hence these equations may be solved for the coefficients of \( R(s) \). If the model is still unstable then \( R(s) \) is constrained to retain two poles of the original system. This procedure is continued until a stable model is obtained.

It is easy to see why this procedure can never fail to produce a stable reduced order model, since in the limit, we may retain \( k \) poles of the system.

From above it is seen that the choice of system poles to be retained in the reduced model is arbitrary. However, it has been found, by repeated applications to many examples, that Padé approximation leads to unstable models if the system step response has a large overshoot in the initial part of the response, which
may be due to the presence of large magnitude poles which are heavily excited. Hence it seems logical that the large magnitude poles of the system should be retained in the reduced model to ensure stability. The following example will illustrate the applicability of the procedure.

**Example 8**

Consider the following 6th order system transfer function

\[ G(s) = \frac{s^5 + 1014s^4 + 14069s^3 + 69140s^2 + 140100s + 100000}{s^6 + 222s^5 + 1454s^4 + 24840s^3 + 1454100s^2 + 2220000s + 1000000} \]

The poles of \( G(s) \) are located at \(-1, -1, -10, -10, -100, -100\).

Using Padé approximation about \( s = 0 \), it is found that the 2nd and 3rd order models are unstable.

The method of this Section was used to derive a stable 2nd order model. Thus \( G(s) \) is expanded into a power series expansion

\[ G(s) = c_0 + c_1 s + c_2 s^2 + \ldots \]

\[
\begin{align*}
c_0 &= 0.100000 \\
c_1 &= -0.081900 \\
c_2 &= 0.105548 \\
c_3 &= -0.125999 \\
c_4 &= 0.146145
\end{align*}
\]

Retaining one pole at \( s = -1 \), in the reduced model, it was found that the model is unstable. While retaining one pole at \( s = -100 \), the following stable 2nd order model is obtained.

\[
[R(s)]_2 = \frac{3.75072s + 7.7247}{s^2 + 100.77247s + 77.247}
\]

The method of this Section was applied to derive a stable 3rd order model. Retention of the pole at \( s = -1 \), lead to an unstable model, while the retention of the pole at \( s = -100 \), lead to the following stable 3rd order model

\[
[R(s)]_3 = \frac{5.3673s^2 + 15.28212s + 19.68001}{s^3 + 103.12032s^2 + 314.0005s + 196.8001}
\]

The step response of \( G(s) \), \([R(s)]_2 \) and \([R(s)]_3 \) are as shown in graph 7.5.
Response

Graph 7.3

- \( G(s) \)
- \( R_2(s) \)
- \( R_3(s) \)
- \( R_4(s) \)
Graph 7.4

Response vs. Time (Secs.)

- System
- Reduced Model
Graph 7.5

- \( G(s) \)
- \([R(s)]_2\)
- \([R(s)]_3\)
Chapter 8
THE REDUCTION OF DISCRETE-TIME SYSTEMS

8.1 Introduction

Most of the methods of reduction have been concerned with the reduction of continuous-time systems. Our purpose in this Chapter is to show how the methods developed in this thesis may be applied to the reduction of high order discrete-time systems.

It has been shown, in Chapter 3, that the continued fraction method of Chen and Shieh [19] is a special case of Padé approximation about \( s = 0 \), and in fact equivalent to the method of moments. Thus in Section 8.2, the Markov parameters and time moments of discrete-time systems are defined. In Section 8.3, continued fraction expansions are used to derive models which are equivalent to equating a number of time-moments of the system and the reduced order model. In Section 8.4, 'biased' reduced order models, in the sense that the reduced order models approximate the initial transient response as well as the steady state response of the system, are derived using continued fraction expansions. Finally in Section 8.5, the method of Chapter 7, which allows the retention of dominant modes of the system in the reduced model, is extended to the reduction of discrete-time systems.

Several illustrative examples are discussed throughout the Chapter.

8.2 Definition of Time-Moments and Markov Parameters of Discrete-Time Systems

A finite dimensional, discrete-time, linear constant dynamical system has a state-vector description of the form

\[
\begin{align*}
\mathbf{x}(k+1) &= \phi \mathbf{x}(k) + B \mathbf{u}(k) \\
\mathbf{y}(k) &= C \mathbf{x}(k)
\end{align*}
\]

(8.2.1)

where \( k \) is an integer, \( \mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^q \) and \( \mathbf{y} \in \mathbb{R}^p. \phi, B, \) and \( C \) have
dimensions compatible with $x$, $u$ and $y$. Throughout the Chapter however we will only consider the case of single input-single output systems. The external description of a linear system may be stated either in the time-domain or in the transform domain.

For the system (8.2.1), the time-domain description is given by the impulse response functions

$$y_k = C\Phi^k B$$  \hspace{1cm} (8.2.2)

The parameters $y_k$ are known as the Markov parameters of the system. Ho and Kalman [43], use these parameters, which can be derived from input/output measurements, to derive a minimal state-vector realization for the system.

The transform domain description of the system is given by the $z$-transform, transfer function

$$G(z) = C(zI-\Phi)^{-1}B$$  \hspace{1cm} (8.2.3)

The $y_k$ parameters are simply the coefficients of the power series expansion of $G(z)$ about $z = \infty$.

In Section 6.2, it has been shown how a system may be minimally realized using its time-moments. The time-moments for discrete-time systems can be defined by making a direct analogy with the definition existing for continuous systems. The weighting factor is chosen as $k^i$ instead of $t^i$. Thus the time-moments $T_i$ ($i = 0, 1, 2, \ldots$) for the system (8.2.1) are given by [17]

$$T_i = \sum_{k=0}^{\infty} y_k k^i = \begin{cases} G(z)_{z = 1} & \text{for } i = 0 \\ (-1)^i \frac{d}{dz} z^i G(z) & \text{for } i > 0 \end{cases}$$  \hspace{1cm} (8.2.4)

For the continuous case, the time-moments are directly related to the coefficients of the power series expansion of the system transfer function about $s = 0$. While for the discrete-time system, it is easy to see that the time-moments are related to the coefficients of the power series expansion of $G(z)$, about $z = 1$. Thus consider the expansion...
8.3 Continued Fraction Expansions about \( z = 1 \)

Let the system transfer function be of the form

\[
G(z) = \frac{A_21 + A_22 z + A_23 z^2 + \ldots + A_{2n} z^{n-1}}{A_{11} + A_{12} z + A_{13} z^2 + \ldots + A_{1,n+1} z^n}
\]  

(8.3.1)

and let

\[
z = v + 1
\]

Then substituting for \( z \) in (8.3.1) gives

\[
G(v) = \frac{B_{21} + B_{22} v + B_{23} v^2 + \ldots + B_{2n} v^{n-1}}{B_{11} + B_{12} v + B_{13} v^2 + \ldots + B_{1,n+1} v^n}
\]  

(8.3.3)

This is then expanded into a C-type continued fraction about \( v = 0 \), of the form
\[ G(v) = \frac{1}{v} \left( \frac{h_1}{h_2} + \frac{v}{h_{2n-1} h_{2n}} \right) \]  

(8.3.4)

where the coefficients \( h_i \) (\( i = 1, 2, \ldots, 2n \)) may be derived by applying the Routh algorithm [98]. Thus given Equation (8.3.3) the Routh array is formed as follows:

\[
\begin{array}{cccccc}
B_{11} & B_{12} & B_{13} & B_{14} & \cdots \\
B_{21} & B_{22} & B_{23} & B_{24} & \cdots \\
B_{31} & B_{32} & B_{33} & \cdots \\
B_{41} & B_{42} & B_{43} & \cdots \\
& & & & \\
& & & & \\
& & & & \\
\end{array}
\]

(8.3.5)

The coefficients \( h_i \) are then given by

\[ h_i = \frac{B_{i,1}}{B_{i+1,1}}, \quad i = 1, 2, \ldots, 2n \]  

(8.3.6)

A reduced model, of order \( k \) say, is obtained by truncating (8.3.4) after the first \( 2k \) terms. Thus the reduced model \( R(z) \) is given by

\[ R(z) = \frac{1}{h_1 + \frac{(z-1)}{h_2 + \cdots + \frac{(z-1)}{h_{2k}}}} \]  

(8.3.7)

Then by inverting (8.3.7), the reduced order transfer function of the model is obtained. It has been shown in Section 3.2 that \( R(v) \) is the \([k-1,k]\) Pade approximant of \( G(v) \). Hence the power series expansion of \( R(z) \), about \( z = 1 \), agrees with that of \( G(z) \) up to, and including, the term in \((z-1)^{2k-1}\). Thus \( R(z) \) fits the first \( 2k \) time-
moments of $G(z)$.

In the analysis above, C-type continued fraction expansions have been used simply because they have been used in the literature [19, 27]. It is easy to show that equivalent reduced order models may be derived using J-type continued fractions, with a saving in computations, especially if applied to the reduction of multivariable systems or systems described in state-vector form.

Another method of deriving $R(z)$ is by computing the $[k-1,k]$ Padé approximant of $G(v)$, and substituting $v = (z-1)$ in the approximant (see Chapter 3). However from a computational point of view, the use of continued fraction expansions is simpler and numerically more efficient.

Example 1

Consider the following system transfer function

$$G(z) = \frac{z^2 - 1.77z + 0.782}{z^3 - 2.65z^2 + 2.535z - 0.684}$$

To derive a second order model, which fits the first four time-moments of $G(z)$, the procedure outlined above is used. Thus substituting $z = v + 1$ in $G(z)$ and expanding into a continued fraction we get

$$G(v) = \frac{1}{0.083333 + \frac{v}{0.757895 + \frac{v}{0.567610 + \frac{v}{0.215828 +}}}}$$

Thus truncating after the first four terms, and substituting $v$ in terms of $z$ and inverting gives

$$R(z) = \frac{0.973722z - 0.880875}{z^2 - 1.79635z + 0.804088}$$

The step responses of $G(z)$ and $R(z)$ were almost the same, and some of the values are shown below.
<table>
<thead>
<tr>
<th>Time (Secs.)</th>
<th>Original System</th>
<th>Reduced Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>3.97341</td>
<td>3.94370</td>
</tr>
<tr>
<td>1.0</td>
<td>6.29800</td>
<td>6.30770</td>
</tr>
<tr>
<td>1.5</td>
<td>7.81151</td>
<td>7.83603</td>
</tr>
<tr>
<td>2.0</td>
<td>8.86587</td>
<td>8.88646</td>
</tr>
<tr>
<td>3.0</td>
<td>10.19550</td>
<td>10.19850</td>
</tr>
<tr>
<td>4.0</td>
<td>10.94190</td>
<td>10.93730</td>
</tr>
<tr>
<td>5.0</td>
<td>11.37390</td>
<td>11.36920</td>
</tr>
<tr>
<td>6.0</td>
<td>11.62480</td>
<td>11.62760</td>
</tr>
</tbody>
</table>

8.4 Continued Fraction Expansions about \( z = 1 \) and \( z = \infty \)

In Chapter 3, Padé approximation was used to derive biased reduced order models, in the sense that the reduced model \( R(s) \), of order \( m \), fits the first \((m-k)\) time-moments and the first \((m+k)\) Markov parameters of the system, for some integer \( k \) \((1 \leq k \leq m)\). Continued fraction expansions about \( s = 0 \) and \( s = \infty \) were also used to derive equivalent biased reduced order models.

It is easy to show that Padé approximation cannot be applied directly to derive biased reduced order models of discrete-time systems, because it would require the expansion of the reduced order model \( R(z) \) about \( z = 1 \), which is rather cumbersome (since the coefficients of \( R(z) \) are unknown). However continued fraction methods may be used for deriving biased models, using expansions about \( z = 1 \) and \( z = \infty \). Thus given the system of Equation (8.3.1), \( G(z) \) is expanded into a continued fraction about \( z = \infty \) for \((m+k)\) terms as follows:

\[
G(z) = \frac{-1}{A_{1,n+1}z^{-1} + \frac{A_{31} + A_{32}z + \ldots + A_{3n}z^{n-1}}{A_{2n} + A_{21}z + \ldots + A_{2n}z^{n}}}
\]

where

\[
A_{3,j} = A_{1,j} - A_{2,j}A_{1,n+1}A_{2,n}^{-1}
\]
continuing for \((k+m)\) terms we get

\[
G(z) = \frac{z^{-1}}{h_1 + \frac{z^{-1}}{h_2 + \ldots + \frac{z^{-1}}{h_{k+m} + G_{k+m}(z)}}}
\]  

(8.4.1)

where

\[
h_1 = \frac{A_{1,n+1}}{A_{2,n}} ; \quad h_2 = \frac{A_{2,n}}{A_{3,n}} ; \quad h_3 = \frac{A_{3,n}}{A_{4,n-1}} ; \quad \text{etc.,}
\]

and the transfer function \(G_{k+m}(z)\) is of the form

\[
G_{k+m}(z) = \frac{A_{k+m+2,1} + A_{k+m+2,2}z + \ldots + A_{k+m+2,n+1}z^{-1}}{2} + n-\frac{(k+m)}{2}
\]

(8.4.2)

where for simplicity of notation \((k+m)\) is assumed even.

Now \(G_{k+m}(z)\) must be expanded about \(z = 1\). Hence let

\[
z = v + 1
\]

(8.4.3)

and substituting in (8.4.2) leads to

\[
G_{k+m}(v) = \frac{B_{k+m+2,1} + B_{k+m+2,2}v + \ldots + B_{k+m+2,n+1}v^{-1}}{2} + n+1-\frac{(k+m)}{2}
\]

(8.4.4)

which is then expanded about \(v = 0\) to give

\[
G_{k+m}(v) = \frac{1}{B_{k+m+1,1} + B_{k+m+1,2}v + \ldots + B_{k+m+1,n+2}v^{-1}}
\]

(8.4.5)

or rewriting

\[
G_{k+m}(z) = \frac{1}{h_{k+m+1} \cdot \frac{z-1}{h_{k+m+2} \cdot \ldots \cdot \frac{z-1}{h_{2n}}}}
\]

(8.4.6)
where
\[ h_i = \frac{B_{i,1}}{B_{i+1,1}}, \quad i = k+m+1, k+m+2, \ldots, 2n \quad (8.4.6) \]
and the \( B_{i,j} \) are computed using Equation (8.3.5). Hence using
(8.4.5), Equation (8.4.1) may be rewritten in the form
\[ G(z) = \frac{-1}{z-1} \]
\[ \frac{-1}{h_1 +} \]
\[ \frac{-1}{h_2 +} \]
\[ \frac{-1}{h_{k+m} +} \frac{1}{h_{k+m+1} +} \frac{1}{h_{k+m+2} +} \cdots \frac{(z-1)}{h_{2n}} \]
A reduced model \( R(z) \) which fits the first \((m+k)\) Markov parameters and the first \((m-k)\) time-moments of \( G(z) \) may be obtained by truncating (8.4.7) after the first \( 2m \) terms. Thus
\[ R(z) = \frac{-1}{z-1} \]
\[ \frac{-1}{h_1 +} \]
\[ \frac{-1}{h_2 +} \]
\[ \frac{-1}{h_{k+m} +} \frac{1}{h_{k+m+1} +} \frac{1}{h_{k+m+2} +} \cdots \frac{(z-1)}{h_{2m}} \]
which when inverted reduces to the transfer function \( R(z) \) of order \( m \).

Although the dynamics of the reduced order model \( R(z) \) are independent of the order in which the continued fraction is expanded, the procedure outlined above (that is, expansion about \( z = \infty \) first) is computationally the most efficient.

\( R(z) \), of Equation (8.4.8), may be shown to fit the first \((m+k)\) Markov parameters and the first \((m-k)\) time-moments of \( G(z) \) by a proof similar to those outlined in Chapter 3. However for the sake of completeness, the proof will be included here.

**Theorem 8.1**

The power series expansion of \( R(z) \), derived from (8.4.7) about
\( z = 1 \) (\( z = \infty \)) agree with that of \( G(z) \) up to, and including the term in \((z-1)^{m-k-1}(z^{-m-k})\).

**Proof**

Equation (8.4.7) may be rewritten in the form

\[
G(z) = \frac{1}{zh_1 + h_2 + \cdots + \frac{1}{zh_{k+m} + \frac{1}{(z-1)h_{k+m+1} + h_{k+m+2} + \cdots + (z-1)h_{2n}}}}
\]

Then using the formula of Equation (3.5.10), Chapter 3, and Equation (8.4.9), let

\[
a_1 = a_2 = \cdots = a_{k+m+1} = 1
\]

\[
a_{k+m+2} = a_{k+m+3} = \cdots = a_{2n} = (z-1)
\]

\[
b_i = zh_i, \quad i = 1, 3, \ldots, k+m-1
\]

\[
b_i = h_i, \quad i = 2, 4, \ldots, k+m, k+m+1, \ldots, 2n.
\]

Hence we have

\[
A_{2m+2}^2 - A_{2m}^2 = (-1)^{2m}(z-1)^{-m-k}h_{2m+2}^2
\]

also

\[
\frac{1}{B_{2m+2}^2 B_{2m}} = \frac{1}{\beta_0 + \beta_1 (z-1) + \beta_2 (z-1)^2 + \cdots} = \sum_{i=0}^{\infty} \alpha_i (z-1)^i
\]

Hence using (8.4.10) and (8.4.11) we get

\[
\frac{A_{2m+2}^2 - A_{2m}}{B_{2m+2}^2 B_{2m}} = \frac{A_{2m+2}^2 B_{2m} - A_{2m} B_{2m+2}}{B_{2m+2}^2 B_{2m}}
\]

\[
= (-1)^{2m} h_{2m+2} (z-1)^{m-k} \sum_{i=0}^{\infty} \alpha_i (z-1)^i
\]

\[
= \sum_{i=0}^{\infty} \gamma_m i (z-1)^{m-k+i}
\]

Hence it follows that the power series expansion, about \( z = 1 \),
of \( \frac{A_{2m+2}}{B_{2m+2}} \) and \( \frac{A_{2m}}{B_{2m}} \) obtained by truncating (8.4.7) agree up to, and including the term in \((z-1)^{m-k-1}\). Equation (8.4.7) may be rewritten in the form

\[
G(z) = \frac{1}{z-1} + \frac{1}{z} + \frac{1}{z-1} h_{k+m} + \frac{1}{z-1} h_{k+m+1} + \frac{1}{z-1} h_{k+m+2} (z-1)^{-1} + \ldots + \frac{1}{z-1} h_{2n} (z-1)^{-1}
\]

(8.4.13)

Again, using the formula of Equation (3.5.10), Chapter 3, and Equation (8.4.13), let

\[
a_1 = a_2 = \ldots = a_{k+m} = z^{-1}
\]

\[
a_{k+m+1} = \ldots = a_{2n} = 1
\]

\[
b_i = h_i, \quad i = 1, 2, \ldots, k+m, k+m+1, k+m+3, \ldots
\]

\[
b_i = h_i (z-1)^{-1}, \quad i = k+m+2, k+m+4, k+m+6, \ldots
\]

Hence we have

\[
A_{2m+2} B_{2m} - A_{2m} B_{2m+2} = (-1)^{2m} h_{2m+2} (z-1)^{-1} z^{-k-m}
\]

\[
= (-1)^{2m} h_{2m+2} \{z^{-1} z^{-2} + \ldots \} z^{-k-m}
\]

\[
= \sum_{i=0}^{\infty} \alpha_i z^{i-k-m-1}
\]

(8.4.14)

We also have

\[
\frac{1}{B_{2m+2} B_{2m}} = \sum_{i=0}^{\infty} \alpha_i z^{i-1}
\]

(8.4.15)

Hence using Equations (8.4.14) and (8.4.15) gives

\[
\frac{A_{2m+2} B_{2m} - A_{2m} B_{2m+2}}{A_{2m+2} B_{2m}} = \sum_{i=0}^{\infty} \theta_{0,1} z^{i-k-m-1+i}
\]

(8.4.16)

Hence it follows that the power series expansions about \( z = \infty \), of
\[ \frac{A_{2m+2}}{B_{2m+2}} \text{ and } \frac{A_{2m}}{B_{2m}} \text{ agree up to the term in } z^{-m-k}. \]

Since, by definition, the original transfer function
\[ G(z) = \frac{A_{2n}}{B_{2n}}, \]
we have for \( 1 \leq m \leq n, \)
\[ G(s) = \frac{A_{2n}}{B_{2n}} = \left( \frac{A_{2n}}{B_{2n}} - \frac{A_{2n-2}}{B_{2n-2}} \right) + \left( \frac{A_{2n-2}}{B_{2n-2}} - \frac{A_{2n-4}}{B_{2n-4}} \right) + \ldots \]
\[ + \left( \frac{A_{2m+2}}{B_{2m+2}} - \frac{A_{2m}}{B_{2m}} \right) \]
\[ = \sum_{i=0}^{\infty} \gamma_{n,i} (z-1)^{n-k+i} + \ldots \sum_{i=0}^{\infty} \gamma_{k,i} (z-1)^{m-k+i} \]
\[ = \sum_{i=0}^{\infty} \gamma'_{m,i} (z-1)^{m-k+i} \]
by using Equation (8.4.12). Similarly by using Equation (8.4.16),
we have
\[ G(s) - \frac{A_{2m}}{B_{2m}} = \sum_{i=0}^{\infty} \gamma''_{m,i} (z-1)^{m-k-1+i} \]
Hence the power series expansion about \( z = 1 \) \((z = \infty)\) of the
reduced order model \( R(z) \), which is equal to \( \frac{A_{2m}}{B_{2m}} \), obtained from
Equation (8.4.7) agree with that of \( G(z) \) up to, and including, the
term \((z-1)^{-m-k}(z^{-m-k})\). Hence \( \Delta \) proof \( \Delta \) completed.

Example 2

This example was chosen to illustrate how continued fraction
expansions may be used to derive biased reduced order models.
Consider the following transfer function
\[ G(z) = \frac{8z^2 - 15.28282224z + 7.31312706}{z^3 - 2.62840539z^2 + 2.30036681z - 0.67031978} \]
The Markov parameters of \( G(z) \) are
\[ \gamma_1 = 8.0000; \quad \gamma_2 = 5.7444; \quad \gamma_3 = 4.0087; \quad \gamma_4 = 2.6849; \ldots \]
while the coefficients of the power series expansion of \( G(z) \), about
\( z = 1 \), are
\[ c_0 = 18.4176; \quad c_1 = -51.8311; \quad c_2 = 2079.4390 \]
\[ c_3 = -54658.4627; \ldots \]
Let the second order model be of the form

\[ R(s) = \frac{a_0 + a_1 z}{b_0 + b_1 z + z^2} \]

Hence there are four unknowns. There are five possible models (of second order).

I. \( R(z) \) to 'fit' the first four time-moments of the system. \( G(z) \) is expanded into a continued fraction about \( z = 1 \) and the expansion is truncated after the first four quotients. For this example the model was found unstable.

II. \( R(z) \) to 'fit' the first three time-moments and the first Markov parameter of the system. \( G(z) \) is expanded about \( z = \infty \) for one term and about \( z = 1 \) for another three terms to give

\[ R_{3,1}(z) = \frac{z^{-1}}{14.14357353 + (z-1) 0.17988635 + (z-1) 0.02715387} \]

which when inverted gives

\[ R_{3,1}(z) = \frac{8.000000z-7.960998}{z^2-1.55966881z+0.56178648} \]

The indices on \( R_{3,1}(z) \) indicate the number of time-moments and number of Markov parameters fitted.

The first Markov parameter of \( R_{3,1}(z) \) is

\[ Y_1 = 6.0 \]

while the first three coefficients of the power series expansion of \( R_{3,1}(z) \), about \( z = 1 \) (which are proportional to the time-moments) are

\[ c_0 = 18.4175; \quad c_1 = 51.8310; \quad c_2 = 2079.4390 \]

which are clearly the same as those of the system (allowing for small computational errors). The step responses of \( G(z) \) and \( R_{3,1}(z) \) were plotted in graph 8.1. As expected the response of \( R_{3,1}(z) \) is a good approximation to that of \( G(z) \) at large time, but rather poor
at small times.

III. $R(z)$ to 'fit' the first two time-moments and the first two Markov parameters of the system. In this case it was found that $R(z)$ was unstable.

IV. $R(z)$ to 'fit' the first time-moment and the first three Markov parameters of the system $G(z)$ is expanded into a continued fraction about $z = \infty$ for three terms and then about $z = 1$ for one term. This is then inverted to give

$$R_{1,3}(z) = \frac{8.0z-7.238796}{z^2-1.62289906z+0.66422182}$$

The first three Markov parameters of $R(z)$ are

$Y_1 = 8.0, \quad Y_2 = 5.7444; \quad Y_3 = 4.0087$

while the first coefficient of the power series expansion of $R_{1,3}(z)$, about $z = 1$, is

$$c_0 = 18.4176$$

which are clearly the same as those of $G(z)$. The step response of $R_{1,3}(z)$ is plotted as shown in graph 8.1. As expected the response at small times is a good approximation to the response of $G(z)$.

V. $R(z)$ to fit the first four Markov parameters of the system. $G(z)$ is expanded into a continued fraction, about $z = \infty$, for four terms and then truncated and inverted to give

$$R_{0,4}(z) = \frac{8z-10.557925}{z^2-2.0377846z+0.96213425}$$

The first four Markov parameters of $R_{0,4}(z)$ were the same as those of the system. However the step response of $R_{0,4}(z)$, although approximating the system very well at small times, was a very bad approximation at large time, as would have been expected.
8.5 Padé Approximation to Allow Retention of Dominant Modes

For discrete-time systems, the modes of the system lie inside the unit circle (assuming the system is stable). The dominant modes of the system are nearest to one (in absolute value), while the negligible modes are nearest the origin. Hence given the high order system transfer function as shown in Equation (8.3.1), it is expanded into a power series about $z = \infty$,

$$G(z) = D_0 + \frac{D_1}{z} + \frac{D_2}{z^2} + \cdots$$

where $D_i$ ($i = 0, 1, 2, \ldots$) are constant scalars. Then applying Koenig's theorem and its generalisation to Equation (8.5.1), we can determine the dominant modes of the system (the largest magnitude poles), say $p_1, p_2, \ldots, p_m$. It should be noted that computationally the convergence of Koenig's theorem may be slow because of the close proximity of the poles of the system in the unit circle. Let the reduced model be given by

$$R(z) = \frac{a_0 + a_1 z + \cdots + a_{m-1} z^{m-1}}{b_0 + b_1 z + \cdots + b_{m-1} z^{m-1} + z^m}$$

where the $b_i$ ($i = 0, 1, 2, \ldots, m-1$) may be computed from $p_i$ ($i = 1, 2, \ldots, m$). The numerator coefficients of $R(z)$ are then determined by constraining $R(z)$ to fit the first $(m-1)$ time-moments of $G(z)$. That is

$$a_0 = b_0 c_0$$

$$a_1 = b_0 c_1 + b_1 c_0$$

$$\cdots \cdots \cdots \cdots$$

$$a_{m-1} = b_0 c_{m-1} + b_1 c_{m-2} + \cdots + b_{m-1} c_0$$

where the $c_i$ ($i = 0, 1, \ldots, m-1$) are as defined in Equation (8.2.5).
Chapter 9
SUMMARY AND CONCLUSIONS

A survey of model reduction techniques and their application in control system design was given. The method of continued fraction synthesis and the time-moments technique were conceptually and computationally the simplest.

For asymptotically stable systems, the method of continued fractions and the time-moments technique were shown to be equivalent. Their relation to the Padé approximation problem was established.

An equivalent method of reduction, based on J-type continued fraction expansions, which is computationally superior was introduced. Further, continued fraction expansions were used to derive biased reduced models. Padé approximation and continued fraction methods were used for the reduction of systems described in state vector form. Computationally the use of J-type continued fractions was found to be the most efficient. These methods have the important property that they may be applied to the reduction of systems described by irrational transfer functions. The methods may also be used to approximate non-linear systems by constant reduced order models.

The above methods were extended to the problem of reducing multivariable systems, with an equal number of inputs and outputs. It was found that if a Padé approximant exists, then computing the reduced model by Padé approximation is computationally easier than using continued fractions. However if a Padé approximant does not exist, then the J-type continued fraction method should be used. Further, Padé approximation was extended to the reduction of a general multivariable system.

Padé approximation and continued fraction methods when applied to the reduction of multivariable systems, necessitate the inversion
of rational and polynomial matrices. An algorithm for the inversion of systems was introduced, which is conceptually simple and computationally easy in comparison with other techniques.

The problem of Padé approximation was shown to be related to the problem of partial minimal realization. This is a very important relation since existing minimal realization algorithms may be used for the reduction in the order of systems, thus alleviating the necessity of writing an algorithm for the reduction process. The algorithms may also be used for the reduction of multivariable systems. An algorithm for minimally realizing a system, described by a set of constant differential equations, was introduced. A method was also introduced for the realization of a system from a combination of time moments and Markov parameters. This is useful in identification and when deriving biased reduced order models.

A new method of reduction was introduced. The method retains the advantages of the Padé approximation methods, and the modal methods which are based on the retention of some of the system modes in the reduced model. The method makes use of Koenig's theorem and its generalisation, which have the important property of approximating a number of poles of the system in one operation. The method will always produce a stable (unstable) reduced order model if the high order system is stable (unstable). This is a property which many of the other methods of reduction lack. Further the method may be used for the reduction of oscillatory systems, for which the Padé approximation techniques may fail to produce a stable model.

An algorithm for ensuring that the reduced order models, derived by Padé approximation methods are stable was introduced. The algorithm is a special case of the new method of reduction (introduced in Chapter 7) and is based on the retention of one or more of the system
poles in the reduced model. The algorithm will always produce a stable reduced model if the high order system is stable.

Finally, the methods outlined in the thesis were extended to the problem of reducing discrete-time systems.

In conclusion, J-type continued fractions should be used for the reduction of single input-single output systems, while Padé approximations, if they exist, should be used for the reduction of multivariable systems. In the case when an algorithm for the minimal realization of systems, based on the Hankel matrix approach, is available, then it may be used for the reduction process.

Padé approximations (about one or more points) are very suitable for the reduction of systems with a pure time delay and for approximating distributed parameter systems.

If it is desired that the reduced model should retain some of the modes of the system, or if the reduced model is required to be stable (unstable) when the high order system is stable (unstable), or if the system to be reduced is oscillatory, then the method of Chapter 7 should be used. It should be noted that for single input-single output systems, the computational requirements of the method is slightly more than that required when using Padé approximation, and in fact an algorithm for Padé approximation may be used to derive the reduced models for the method of Chapter 7. For the reduction of multivariable systems, the method of Chapter 7 is more versatile than that of Chapter 1, and also the computational requirements are less.

All of the methods are of general applicability, that is they can be applied to systems described in state-vector form, by a transfer function or by a set of input/output measurements. This is a property that many of the existing methods of reduction lack.

Throughout the thesis, the various algorithms have been
illustrated by numerical examples, some of which had been borrowed from the literature. These examples were chosen to bring out certain features of the various methods. We have not applied these algorithms to any specific physical system, but the principal of equating moments and Markov parameters has been used for modelling physical systems by a number of authors [58, 81, 82].

An outstanding problem in the use of Padé approximation is the problem that the reduced model may be stable (unstable) even though the system is unstable (stable). It is suggested that further research may concentrate on this problem. Although we have suggested a method for overcoming the problem, it would be very useful, if it can be stated prior to reduction, whether a reduced model of certain order will be stable or not.

As has been shown in Chapter 2, a large number of methods have been developed for the reduction of high order systems. Therefore it is felt that future research efforts should concentrate more on the applications of model reduction in the design of control systems, than on the development of new methods of reduction. One such problem is the design of feedback controllers, using reduced models. If an optimal controller is derived for the reduced model, it may be found that by applying this controller to the original system, the whole system becomes unstable.
REFERENCES


36. Gerth, W.; On the construction of low order linear state-variable models from measured data, 2nd IFAC Symposium on Multivariable Control Systems, Dusseldorf, Germany, paper 1.2.1, 12 pages, October 1971.

37. Geugen, C.J.; An algebraic algorithm for reducing a differential system to a state form, 2nd IFAC Symposium on Multivariable Control Systems, Dusseldorf, Germany, paper 1.2.2, 11 pages, October 1971.


46. Hsieh, T.C.; A method for linear system simplification, Proc. 13th


76. Rissanen, J.; Recursive evaluation of Pade approximants for matrix sequences, IBM Research Laboratory, San Jose, California, 1972.


79. Roberts, P.D. and Dunham, B.P.; An investigation into the
dynamical reduction of large multivariable systems, Imperial
Chemical Industries Ltd., C.I.R.L. Research Note BPD/PDR/69/
10, 1969.

80. Rogers, R.O. and Sworder, D.D.; Suboptimal control of linear
systems derived from models of lower order, Proc. Joint

81. Rossen, R.H. and Lapidus, L.; Minimum realization and systems
modelling: I. Fundamental theory and algorithms, A.I.Ch.E.J.,

82. Rossen, R.H. and Lapidus, L.; Minimum realization and systems
modelling: II. Theoretical and numerical extensions, A.I.Ch.E.J.,

83. Sain, M.K. and Massey, J.L.; Invertibility of linear time-
AC-14, pp.141-9, 1969.

84. Sanathan, C.K. and Koerner, J.; Transfer function synthesis

85. Shamash, Y.; Approximation of linear time-invariant systems,
Proc. Conf. on Padé Approximants and their Applications, Ed.

86. Shamash, Y.; Continued fraction methods for the reduction
of linear time-invariant systems, IEE Conf. on Computer Aided

87. Shamash, Y.; Minimal realization of differential systems,
International Conference On Systems And Control, PSG College
of Technology, Coimbatore-641004, India, August 30-September 1


89. Silverman, L.M.; Realization of linear dynamical systems, IEEE


91. Sinha, N.K. and De Bruin, H.; Application of low-order models
for near optimal control of high order systems, Int. J. Control,

92. Sumner, A.; A method of fitting rational transfer functions

93. Tether, A.J.; Construction of minimal linear state-variable
models from finite input/output data, Proc. Joint Automatic


101. Wolovich, W.A.; *The determination of state space representation of linear multivariable systems*, 2nd IFAC Symp. on Multivariable Systems, Dusseldorf, Germany, paper 1.2.3., October 1971.


APPENDIX I

Elementary Properties of Continued Fractions

In this Appendix, definitions and some properties of continued fractions are discussed. A very thorough exposition of the theory of continued fractions may be found in the excellent book by Wall [98].

The expression

\[ \frac{a_1}{b_1 + \frac{a_2}{b_2 + \frac{a_3}{b_3 + \ldots + \frac{a_n}{b_n}}}} \]

where \( a_i, b_i \) may be real or complex numbers, is called a continued fraction. It is the \( n \)th convergent of the infinite continued fraction.

\[ \frac{a_1}{b_1 + \frac{a_2}{b_2 + \frac{a_3}{b_3 + \ldots + \frac{a_n}{b_n + \frac{c_{n+1}}{b_{n+1}}}}}} \]

The \( n \)th convergent of the infinite continued fraction may be written in the form

\[ \frac{A_n}{B_n} = \frac{a_1}{b_1 + \frac{a_2}{b_2 + \frac{a_3}{b_3 + \ldots + \frac{a_n}{b_n}}}} \]

where the \( A_n \) and \( B_n \) are polynomials in \( a_i, b_i \) (\( i = 1, \ldots, n \)).

The \( a_k \) is called the \( k \)th partial numerator.

The \( b_k \) is called the \( k \)th partial denominator.
The ratio \( \frac{a_k}{b_k} \) is called the \( k^{th} \) partial quotient.

The ratio \( \frac{A_k}{B_k} \) is called the \( k^{th} \) convergent or the \( k^{th} \) approximant.

The quantities \( A_k \), \( B_k \) may be computed by means of the fundamental recurrence formulae

\[
\begin{align*}
A_0 &= 0, \quad B_0 = 1 \\
A_1 &= a_1, \quad B_1 = b_1 \\
A_k &= b_k A_{k-1} + a_k A_{k-2} \\
B_k &= b_k B_{k-1} + a_k B_{k-2}
\end{align*}
\]  

(1.3)  

(1.4)

\( A_k \) is called the \( k^{th} \) numerator and \( B_k \) the \( k^{th} \) denominator of the continued fraction (I.1).

Two continued fractions are said to be equivalent if, and only if, all their respective approximants \( \frac{A_n}{B_n} \) are equal.

The even part of a continued fraction is defined as the continued fraction whose sequence of approximants \( \frac{A'_k}{B'_k} \) (\( k = 1, 2, 3, 4, \ldots \)) is the sequence of even approximants \( \frac{A_n}{B_n} \) (\( n = 2, 4, 6, \ldots \)) of the given continued fraction. That is

\[
\begin{align*}
\frac{A'_1}{B'_1} &= \frac{A_2}{B_2} = \frac{A_4}{B_4} = \ldots \\
\frac{A'_k}{B'_k} &= \frac{A_{2k}}{B_{2k}} = \ldots
\end{align*}
\]

Similarly, the odd part of a continued fraction is defined as the continued fraction whose sequence of approximants \( \frac{A''_k}{B''_k} \) (\( k = 1, 2, \ldots \)) is the sequence of odd approximants \( \frac{A_n}{B_n} \) (\( n = 1, 3, 5, 7, \ldots \)) of the given continued fraction. That is

\[
\begin{align*}
\frac{A''_1}{B''_1} &= \frac{A_1}{B_1} = \frac{A_3}{B_3} = \ldots \\
\frac{A''_k}{B''_k} &= \frac{A_{2k-1}}{B_{2k-1}} = \ldots
\end{align*}
\]

Thus consider the following example

\[
\frac{1}{a_2} \quad \frac{1}{1+a_3} \quad \frac{1}{1+a_4} \quad \ldots
\]  

(I.5)
Then it can be shown [98] that the even part of (I.5) is of the form

\[
\frac{1}{1 + a_2 + \frac{a_3 a_4}{1 + a_4 + \frac{a_5 a_6}{1 + a_6 + \ddots}}}
\]

and its odd part is given by

\[
\frac{a_2}{1 + a_2 + a_3 + \frac{a_4 a_5}{1 + a_4 + a_5 + \ddots}}
\]

**S-Fractions (C-Fractions)**

These are continued fractions of the form

\[
\frac{c_1}{1 + \frac{c_2 z}{1 + \frac{c_3 z}{1 + \ddots}}}
\]

where \( z \) is a complex variable. The approximants of (I.8) are rational functions of \( z \), and \( A_k(z), B_k(z) \) are polynomials in \( z \). It can be shown [98] that \( A_{2n}(z), B_{2n}(z), A_{2n+1}(z) \) and \( B_{2n+1}(z) \) are of degree \((n-1), n, n \) and \( n \) respectively.

The continued fraction expansion given by

\[
G(z) = \frac{c_1}{1 + \frac{c_2 z}{1 + \frac{c_3 z}{1 + \ddots}}}
\]

will terminate if, and only if, the function \( G(z) \), being expanded is a rational function.

The continued fraction (I.8) may be rearranged and written in the form
which is known as the Second Cauer Form. This form is widely used in circuit synthesis, for the construction of ladder networks. Throughout the thesis, continued fractions of the form (I.8) and (I.9) will be referred to as C-type continued fractions.

**J-Fractions (Jordan-type Continued Fractions)**

These are continued fractions of the form

\[
\frac{1}{a_1z+b_1 + \frac{1}{a_2z+b_2 + \frac{1}{a_3z+b_3 + \cdots}}} \tag{I.10}
\]

where \( z \) is a complex variable. The approximants of (I.10) are rational functions of \( z \), and \( A_k(z) \), \( B_k(z) \) are polynomials in \( z \).

It can be shown [98] that the polynomials \( A_n(z) \), \( B_n(z) \), \( A_{n+1}(z) \) and \( B_{n+1}(z) \) are of degree \((n-1)\), \( n \), \( n \), and \((n+1)\) respectively.

The continued fraction (I.10) may be rearranged and written in the form

\[
\frac{z^{-1}}{a_1z^{-1} + \frac{z^{-2}}{a_2z^{-1} + \frac{z^{-2}}{a_3z^{-1} + \cdots}}} \tag{I.11}
\]

which is the form used in this thesis (with \( s = z^{-1} \)). Marshal [61] used the continued fraction (I.10) to derive a state-vector representation of a ladder network, with the state matrix being tridiagonal.

Throughout the thesis, continued fractions of the form (I.10)
Continued Fraction Inversion

The inversion problem is that, given a finite continued fraction whose elements are known, what is the corresponding rational function. Chen and Shieh [21] claim that the inversion problem is much more difficult and tedious than the expansion problem. This is not the case, and an algorithm for the inversion problem which is much easier to compute than that of reference [21] is now given. The algorithm is based on the fundamental recurrence formulae (I.3) and (I.4).

Suppose we are given the continued fraction

\[ G(z) = \frac{1}{a_1 z + b_1 + \frac{1}{a_2 z + b_2 + \cdots + \frac{1}{a_n z + b_n}}} \]  

which is to be inverted. Let the coefficients of the denominator (numerator) of the approximant of \( G(z) \) be \( k_{i,j} \), where

- \( i \) = the order of the denominator (numerator)
- \( j \) = the power of \( z \), \( j \leq i \)

Thus for example, \( k_{3,2} \) would refer to the coefficient of the \( z^2 \) term in the denominator (numerator) of the 3rd order approximant.

The \( k_{i,j} \) for the denominator of an \( n \)th order approximant can be calculated by using the following recursive formula (which is derived from (I.3) and (I.4))

\[ k_{n,j} = b_n k_{n-1,j} + a_n k_{n-1,j-1} + k_{n-2,j-2} \]  

where

- \( k_{i,j} = 0 \) \( \forall (j = i, i \geq 0) \), and \( \forall j < 0 \)
- \( k_{i,j} = 0 \) \( \forall i < 0 \)

and where the \( a_i \) and \( b_i \) are as defined in Equation (I.12). The
elements of the numerator may be computed similarly.

Relation (I.12) has been derived for a J-type continued fraction. A similar relation may be derived for C-type continued fractions by using the fundamental recurrence formula of Equations (I.3) and (I.4).

APPENDIX II

Pade Approximations and Their Properties

Definition

A Pade approximant is a rational function \( \frac{P_m(x)}{Q_n(x)} \) where \( P_m(x) \) and \( Q_n(x) \) are polynomials, in \( x \), of degrees \( m \) and \( n \) respectively, and is denoted symbolically by \([m,n]\). The rational function \([m,n]\) is said to be a Pade approximation of the function \( f(x) \), if, and only if the power series expansion of \([m,n]\) is identical with that of \( f(x) \) up to, and including terms of order \( x^{m+n} \). Further, the order of the approximant is said to be \( N = (m+n) \), since there is agreement between the first \( n+m \) terms of the power series expansion of the approximant and the function \( f(x) \).

Associated with the Pade approximations is a geometrical configuration, known as the Pade table, as shown in Figure II.1 below.

```
Fig. II.1 The Pade Table
```

\[ [0,0] \quad [0,1] \quad [0,2] \quad ... \\
\[ [1,0] \quad [1,1] \quad [1,2] \quad ... \\
\[ [2,0] \quad [2,1] \quad [2,2] \quad ... \\
\vdots \quad \vdots \quad \vdots \\
\]
It consists of a doubly infinite array for which the entry in position \([m,n]\) is a polynomial of degree \(m\) divided by a polynomial of degree \(n\).

Let the function to be approximated be defined by the power series

\[
f(x) = c_0 + c_1 x + c_2 x^2 + \ldots
\]

and let the Padé approximant be defined by

\[
\frac{P_m(x)}{Q_n(x)} = \frac{a_0 + a_1 x + a_2 x^2 + \ldots + a_m x^m}{b_0 + b_1 x + b_2 x^2 + \ldots + b_n x^n}
\]

Now since the power series expansion of Equation (II.2) is to agree with that of Equation (II.1) as far as, and including the term in \(x^{m+n}\), that is the power series

\[
Q_n(x)f(x) - P_m(x)
\]

should commence with a term in \(x^{m+n+1}\). This leads to the following set of linear simultaneous equations

\[
a_0 = b_0 c_0 \\
a_1 = b_0 c_1 + b_1 c_0 \\
\vdots \\
a_m = b_0 c_m + b_1 c_{m-1} + \ldots + b_m c_0 \\
\vdots \\
0 = b_0 c_{m+n} + b_1 c_{m+n-1} + \ldots + b_n c_m
\]

which serves uniquely to derive the coefficients of (II.2). It should be noted that either \(b_n\) or \(b_0\) are taken to be equal to unity.

The condition for a Padé table of the function \(f(x)\) to be constructed is that all the Hankel determinants \([98]\]

\[
P^n_0 = \begin{vmatrix}
c_0 & c_1 & \ldots & c_{n-1} \\
c_1 & c_2 & \ldots & c_n \\
\vdots & \vdots & \ddots & \vdots \\
c_{n-1} & c_n & \ldots & c_{2n-2}
\end{vmatrix}
\]

should be non-zero. In this case, the Padé table is said to be \textit{NORMAL}. 
In reference [45], Householder introduced a convenient formulation for determining the Padé approximants of Equation (II.1), which is as follows: let

\[ f_n(x) = c_0 + c_1 x + c_2 x^2 + ... + c_n x^n \]  

(II.5)

Define, the two sets of determinants

\[
N_{v,p}(x) = \begin{vmatrix}
  c_{v-p+1} & c_{v-p+2} & \cdots & c_v & f_{v-p}(x) \\
  c_{v-p+2} & c_{v-p+3} & \cdots & c_{v+1} & f_{v-p+1}(x) \\
  \vdots & \vdots & \ddots & \vdots & \vdots \\
  c_{v+1} & c_{v+2} & \cdots & c_{v+p} & f_{v}(x) \\
\end{vmatrix} 
\]  

(II.6)

\[
K_{v,p}(x) = \begin{vmatrix}
  c_{v-p+1} & c_{v-p+2} & \cdots & c_v & x^p \\
  c_{v-p+2} & c_{v-p+3} & \cdots & c_{v+1} & x^{p-1} \\
  \vdots & \vdots & \ddots & \vdots & \vdots \\
  c_{v+1} & c_{v+2} & \cdots & c_{v+p} & 1 \\
\end{vmatrix} 
\]  

(II.7)

**Theorem II.1**

If \( N_{v,p}(x) \) and \( K_{v,p}(x) \) do not vanish identically, then \( \frac{N_{v,p}(x)}{K_{v,p}(x)} \) is the \([v,p]\) entry in the Padé table, for \( f(x) \). Moreover, an entry in the table if it exists, is unique up to a constant multiplier in numerator and denominator.

The proof of this theorem is very simple and may be found on pages 57-8, in reference [45].

The minors of the determinant (II.7) may be used for testing the stability of the reduced order models derived in Chapter 3. It is not suggested that (II.6) and (II.7) should be used for computing these Padé approximants, since they would be computationally wasteful.

It may be shown that the Padé approximant \([m,n]\) of the inverse of a function is the inverse of the \([n,m]\) approximant of the function. That is, if we denote the \([m,n]\) Padé approximant of \( f(x) \) by \( f_{[m,n]}(x) \),
then it can be shown that
\[ f^{[n,m]}(x)^{-1} = f(x)^{-1}. \]

The last \( n \)-equations of (II.3) may be solved for the coefficients \( b_i \) \((i = 0, 1, \ldots, n-1)\). This would involve the inversion of an \((nxn)\) matrix. Then the first \( m \) equations may be used to compute the \( a_i \) \((i = 0, 1, \ldots, m)\). However for computational purposes, if \( m \) is less than \( n \), then the \([n,m]\) Padé approximant of \( f^{-1}(x) \), where \( f(x) \) is a given function, should be computed using Equation (II.8). In this case, the denominator of the \([n,m]\) Padé approximant is of order \( m \), and hence Equation (II.8) will only require the inversion of an \((mxm)\) matrix.

**Pade Approximations about two (or more) Points**

The generalisation of the Padé approximant method about two (or more) points was first introduced by Baker et. al. [10]. The idea is quite simple. Sometimes, information about the function to be approximated is available at two (or more) points. They suggest that this additional information about the function should be taken into account by requiring the Padé approximants to satisfy, exactly, the conditions at the other points as well as those at the origin.

Baker et. al. [10] imposed the value of the function at infinity on the Padé approximants. The required modification in the linear equations, Equation (II.3), which determine the coefficients of the approximant are very simple. The equation which makes the last power series coefficient of the function and its approximant equal, is replaced by one that makes the Padé approximant equal to a given value of the function at infinity.

A number of papers have been written about Padé approximations, which have been found very useful in providing quantitative information about the solution of many problems of physics and chemistry [9, 11, 85].