General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

(NASA-CR-170100) ALTERNATIVES FOR JET ENGINE CONTROL Annual Technical Progress Report, 1 Oct. 1981 - 30 Sep. 1982 (Notre Dame Univ.) 197 p HC A09/MF A01 CSCL 21E

N83-20945

Unclas
G3/07 03090

MAR 1983
RECEIVED
NASA STI FACILITY
ACCESS DEPT.

Department of

ELECTRICAL ENGINEERING

UNIVERSITY OF NOTRE DAME, NOTRE DAME, INDIANA

Twelve Month Technical Progress Report

to the

National Aeronautics and Space Administration

on

NASA Grant NSG-3048*

ALTERNATIVES FOR JET ENGINE CONTROL

October 1, 1981 - September 30, 1982

*Under the Direction of

Dr. Michael K. Sain
Department of Electrical Engineering
University of Notre Dame
Notre Dame, Indiana 46556

ABSTRACT

This report deals with progress made on the Grant NSG-3048 during the twelve month period beginning October 1, 1981 and ending September 30, 1982. The NASA Technical Officer for this period was Dr. Kurt Seldner of Lewis Research Center. The director of the research at the University of Notre Dame was Dr. Michael K. Sain, who has been assisted by Mr. Stephen Yurkovich, a fellow in the Department of Electrical Engineering, by Mr. Joe P. Hill, and by Mr. Thomas A. Klingler, research assistants, in the Department of Electrical Engineering. Mr. Hill received the degree of Master of Science during this period, for his June 1982 thesis entitled "Solution of Nonlinear Optimal Control Problems Using the Algebraic Tensor: An Example". Mr. Klingler expects to complete research investigations for the Master of Science degree very shortly. Mr. Yurkovich may complete requirements for the degree of Doctor of Philosophy in 1983.

Researches during the preceding calendar year have centered on basic topics in the modeling and feedback control of nonlinear dynamical systems. Of special interest have been the following topics: (1) the development of models of tensor type for a digital simulation of the QCSE gas turbine engine; (2) the extension, to nonlinear multivariable control system design, of the concepts of total synthesis which trace their roots back to certain early investigations under this grant; (3) the role of series descriptions as they relate to questions of scheduling in the control of gas turbine engines; (4) the development of computer-aided design software for tensor modeling calculations; (5) further enhancement of the softwares for linear total synthesis, mentioned above; and (6) calcula-

tion of the first known examples using tensors for nonlinear feedback control.

A number of major milestones have occurred during this year of study. Most crucial has been the steady progress of the computer software needed to perform tensor modeling and simulation. The advance of this code is now making it possible to begin a more systematic examination of tensor model identification and order reduction. Increasing availability of this capability has made it easier to determine certain of the key tradeoffs involved with use of tensor models, as for example their increased dynamical quality versus their useful region. In what may be the most significant theoretical development of the year's activity, work is underway to evaluate the effects of redefining the groups upon which, and into which, nonlinear maps act. For interesting cases, the groups can be redefined in such a manner that the nonlinear maps become linear. Among the results already following from this discovery are the definition of a variety of new nonlinear sensitivity functions, of the comparison type. The same idea, in a different application, has permitted the definition of a nonlinear feedback synthesis problem. Finally, a complete calculation has been carried out for the feedback tensors in a nonlinear control example. This calculation has been of the utmost importance in setting goals for the type of software which will be needed for general feedback control, with tensors. Preliminary steps to plan such software have been set in motion.

ACKNOWLEDGMENTS

The funded research on this grant has been aided by the voluntary, unfunded efforts of a number of individuals. We would like to thank Mr. Joseph A. O'Sullivan and Mr. Leo McWilliams in this regard. Mr. O'Sullivan, who participated as a senior, has continued on as a graduate research assistant. Mr. McWilliams is a graduate student in electrical engineering and is doing research with the help of the Minorities Consortium.

Special thanks are due to Dr. R. Michael Schafer, who has been most helpful in regard to issues concerning the PDP-11 computer, and to Mr. Joe P. Hill, whose M.S. Thesis forms the core of this report.

We also acknowledge encouragement and support extended to the project by the Department of Electrical Engineering.

Finally, we are pleased to thank Mrs. T. Youngs, who has prepared the typescripts.

TABLE OF CONTENTS

I. INTRODUCTION

In this report, we discuss progress which has been made on NASA Grant NSG-3048, entitled "Alternatives for Jet Engine Control", during the twelve month period beginning on October 1, 1981 and ending on September 30, 1982.

1.1. Overview of Report

The report is organized broadly into a main body, consisting of six sections,and seven appendices. The principal portion of the main body involves Sections II, III, and IV, which deal with the explicit tensor calculations involved in a substantial example of nonlinear feedback control, based upon tensor system models developed in an earlier year of work under this grant. These sections are supported by Section 1.2, which gives certain literature background, by Section VI, which contains the references, and by Appendix G, which illustrates, in an introductory way, typical calculations which become involved.

Insofar as we know, this substantial example is the first of its type to be completed. More comment on the insights which follow from the main body of the report are provided in Section V, where additional discussion is directed to the appendices.

Appendix E, "A Computer-Aided Design Package for Nonlinear Model Applications", is a report on the crucial software developments which have been, and continue to be, the backbone of studies in tensor modeling and simulation. Steady progress in this regard is now making possible further work in tensor model identification and order reduction. A complete description of this software is planned for the next grant period. Appendix B, "An Application of Tensor Ideas to Nonlinear Modeling of a

Turbofan Jet Engine", illustrates the potential of this developing software for work with engine simulations, such as the QCSEE.

Appendix D, "Controller Scheduling: A Possible Algebraic Viewpoint", continues the investigation of algebraic frameworks which may capture the essence of the practical work in control schedules. It is believed that these studies may hold one of the keys to reducing gaps between the theory of nonlinear control systems and its application. In particular, it is planned to use ideas which grow out of this work as a guide to resolving the tradeoffs between increased dynamical quality in nonlinear models and their region of validity. More study is needed in this area.

Sections C and F deal with material on the Total Synthesis Problem of multivariable control. In the linear case, this problem traces a part of its early roots back to studies supported by this grant. It is a problem of feedback synthesis, which has now developed quite a bibliography. Appendix F gives a list of that bibliography, as of the date of the First American Control Conference last year. Item 18 in Appendix F refers to the software associated with this effort, which is ongoing. Though not yet ready for distribution, the software has resulted in several requests for copies, and it is hoped that limited distribution might not be an event too far in the future. Appendix C, "Nonlinear Multivariable Design by Total Synthesis", is a part of the effort to extend to the nonlinear case. Of particular interest in Appendix C is the modification of group structure and scalar multiplication structure on the vector spaces of inputs and outputs for certain nonlinear systems. It is an amazing fact that, in interesting, nontrivial case, such modifications can result in the system becoming linear. The idea is akin to the choice of a special coordinate system which fits the geometry of a physical problem. We have this idea under intense study, and it seems to hold promise in problems of order reduction and identification. If the work proceeds as planned, more results in this regard should become available in next year's report.

Appendix A contains a chronological bibliography of work undertaken in relation to the grant.

With these introductory remarks, we turn next to the main body of the report.

1.2 Remarks on Multilinear Feedback

One subject of this report is the application of concepts from tensor algebra to the generation of optimal feedback controllers for nonlinear dynamical systems. The primary motivation for a study of this topic came from the results presented in [1], where the local theoretical problem was essentially solved. The importance of this work should be emphasized. Although there were works that were previously existing that used multilinear algebra to study series solutions for a response of a nonlinear system, this was the first application of the ideas in tensor algebra to an optimal control problem. Thus, we find that the topics of multilinear algebra and optimization had been extensively studied, but independently of one another.

The research project has received numerous benefits from this particular study. First, we have completed a rather thorough example study of the methods in [1]. We believe that this example study is the only one of its kind in the literature to date. The details of the example are given in Section IV. Second, we have carried out the study without the explicit use of the methods of dual spaces and symmetric tensor algebra. The purpose of doing so is to gain insight concerning the exact role of these two ideas in the nonlinear feedback problem. This was a revealing experience; and much insight has been gained. Third, we have an initial step in constructing software for such feedback calculations. The importance of such softwares can scarcely be overestimated.

The methods of optimal control have, in certain cases, interesting relationships with the methods of stability theory. Because of this, we have reason to believe that these results may assist in model region design.

ORIGINAL PAGE IS
OF POOR QUALITY

The treatment of optimal control problems in the history of the literature has been quite extensive, with much emphasis placed on the so called Linear Quadratic or LQ problem. Broadly speaking, an LQ control formulation consists of a finite-dimensional linear discrete - or continuous-time dynamic system which is to be controlled in such a way as to minimize the value of a performance criterion which is the integral, or sum, of quadratic functions of the system state and control variables plus a quadratic function of the state at some terminal time, t_1. This concept provides the following system description:

$$\dot{x}(t) = A(t)\ x(t) + B(t)\ u(t)\ , \tag{1.1}$$

$$J = M(x(t_1)) + \int_{t_o}^{t_1} L(x,u,t)\ dt\ , \tag{1.2}$$

$$L(x,u,t) = x^T\ Q(t)x + u^T\ R(t)u\ . \tag{1.3a}$$

$$M(x(t_1)) = x^T(t_1)M\ x(t_1)\ . \tag{1.3b}$$

Here, we assume that M is symmetric and positive definite, $R(t)$ is symmetric and positive definite, and $Q(t)$ is symmetric and positive semidefinite. So, given the linear system in (1.1) and the cost functional in (1.2) satisfying the symmetry and definiteness requirements, we wish to find the optimal control, that is, the control which will drive the system so as to minimize the cost functional. We do not go into detail concerning conditions for existence and uniqueness. Basically, the solution of the state regulation problem leads to an optimal feedback system with the property that the components of the state vector $x(t)$ are kept near zero without excessive expenditure of control energy, which is, in essence, the minimization of the cost functional. Thus, given an

ORIGINAL PAGE IS
OF POOR QUALITY

initial perturbation which is usually in the form of the initial state vector $x(0)$, we find that an optimal control should drive the state vectors to zero while simultaneously minimizing the selected performance index. The notion of desiring the state vector near zero arises from the fact that the state variable is defined as an error term [1] which measures deviation from the global trajectory. The fact that this perturbation is required to be sufficiently small allows for the traditional Taylor's series expansion form. It is well known from the literature that we may construct this optimal control as

$$u(t) = K_1(t)\ x(t)\ , \tag{1.4}$$

where

$$K_1(t) = -\ R^{-1}(t)\ B^T(t)\ V(t)\ . \tag{1.5}$$

The matrix $V(t)$ is the solution to the well known matrix Riccati Equation

$$\dot{V}(t) = -V(t)\ A(t) - A^T(t)\ V(t) + V(t)\ B(t)\ R^{-1}(t)\ B^T(t)\ V(t) - Q(t) \tag{1.6}$$

which can be readily solved on the digital computer by integrating numerically backwards in time from the boundary condition

$$V(t_1) = M\ . \tag{1.7}$$

The problem that we wish to consider in this report is more complex, although we will be able to treat the LQ problem as a special case. Namely, we wish to assume that we are given a nonlinear system rather than a linear system and examine the methods necessary to generate higher order controller terms. This control problem is commonly described in the literature as being approximately optimal, or sub-optimal. When considering the synthesis of a control function for a typical nonlinear system, we will find that an infinite number of terms would be necessary in order to formulate a true optimal controller, that is, we must truncate to a finite number of terms and obtain a performance index that is non-minimal. The amount of work that has been done in the generation of suboptimal control algorithms is quite extensive. Therefore, in order to present an account of the research that has been done, we must be succinct and mention only a few of these works. One of the first attempts to formulate a description of the optimal control problem as applied to nonlinear systems is found in [2], where extensive use is made of the methods of Lyapunov and Chetaev. Here, a formal recursive procedure is developed to construct a suboptimal control as a function of a power series in the states. The work of Lukes [3] extended the concepts introduced by Al'brekht and provided useful results pertaining to the existence and uniqueness of an optimal feedback controller. In [4], we find useful applications of the Hamilton-Jacobi-Bellman approach to a number of illustrative examples, as well as a comparison of various techniques that can be used to generate higher order controllers. Additional results and examples of this method were provided in [5] with particular emphasis on the convergence of the procedure. Also, a method for estimating the degradation in performance caused by the truncation of terms in the controller series was presented. We also note the results presented in [6], where linear, second order, and third order controller expressions were produced using the methods presented in [7]. The improved response obtained whenever higher order controller terms are considered provides sufficient motivation to seek another method by which these terms may be obtained. This method, of course, is the tensor algebra. A few of the ideas surrounding the general topic of algebraic system theory have previously been applied to problems in optimality, system modeling, and multivariable feedback loop closures. The latter point was extensively studied in [8], where emphasis was placed on the exterior or skew-symmetric algebra. The most pertinent work was of course [1] which, as was previously mentioned, provided much of the motivation for this report. Another application of the ideas of tensor algebra to systems problems was in [9], which relied heavily on the series expansion concept expressed via the tensor product and applied to the basic system description. Particular emphasis was placed on the subject of nonlinear system modeling, with examples of both homogeneous and nonhomogeneous modeling. These methods proved to be quite effective, yielding much improvement over the standard linear approximations that are typically used in a nonlinear system for modeling purposes. So basically, there would seem to be strong motivation for the use of modern algebra in nonlinear systems and control problems. Much of what we actually do as systems engineers evolves from modern algebraic concepts [10]. Since the typical engineer has had little or no exposure to the concepts of modern algebra, we will outline these concepts as they are related to the solution of our problem. As was noted in [8], modern algebra frequently provides sufficient algebraic framework within which to obtain solutions to systems problems with considerably less effort as compared to other conventional methods. We will find that the tensor algebra viewpoint provides a useful means of expanding a nonlinear system in terms of vectors and matrices, which are expressed via the tensor product.

In Section II we provide some useful mathematical background, with particular emphasis on the topic of tensor algebra. We examine the properties of multilinear mappings and also provide a brief introduction to the symmetric tensor algebra structure, which will be useful when differentiating the tensor product. In Section III we present certain systems concepts, which are basically in the form of series expansions, and then derive the necessary results for the generation of the optimal control terms. The problem formulation for the LQ optimization can be recognized as a special case of the equations where higher order terms are included. Finally, Section IV provides the application of the results derived in the previous chapter to a formulated example problem.

II. SELECTED ALGEBRAIC BACKGROUND

The basic purpose of this section is to provide the reader with the necessary concepts from the subject of tensor algebra, which is the main vehicle that is used to analyze the control problem. It is realized that most readers have not had previous dealings with the somewhat theoretical concept of tensor algebra; therefore the treatment of this subject will not assume any previous knowledge of the topic. The main feature of the algebraic tensor involves the way that it gives ground on dimensionality in order to gain the powerful advantage of linearity.

We begin this section with the basic concepts of multilinear mappings and the properties that allow us to express these multilinear mappings in terms of linear mappings and tensor products. Next, we examine the tensor product of linear mappings and develop the associated Kronecker product and a few of its properties, which will be useful in later derivations. The next section deals with the symmetric tensor algebra, which will be of great importance whenever we consider the possibilities of differentiating the tensor product, a topic that is considered in the final portion of this section. Specifically, we examine the partial derivative problem as related to the tensor product of various copies of the state variable x and the control function u. This problem is inherently related to the minimization operation that will be studied in Section III.

2.1 <u>Multilinear Mappings</u> [1,10,11]

Since many of the ideas surrounding tensor algebra are based on the

ORIGINAL PAGE IS
OF POOR QUALITY

theory of bilinear and multilinear mappings, we begin with a general definition of these mappings. Generally speaking, a multilinear function is a function of vectors that is linear with respect to each vector variable when the others are held constant. This means that

$$\psi(x_1,\ldots,\alpha x_i + \beta y_i,\ldots,x_m) = \alpha\psi(x_1,\ldots,x_i,\ldots,x_m) + \beta\psi(x_1,\ldots,y_i,\ldots,x_m) \,, \tag{2.1}$$

where V_i and U are vector spaces over a field R; x_i, $y_i \in V_i$; $\alpha,\beta \in R$; and $\psi : V_1 \times V_2 \times \ldots \times V_m \to U$. The multilinear mapping ψ has an image which is not, in general, a subspace of U. A simple counterexample is offered in order to illustrate this point.

We let $V_1 = V_2$ be two dimensional spaces with the basis $\{e_1,e_2\}$ and U be a four dimensional space with the basis $\{f_1,f_2,f_3,f_4\}$. Let $x = x_1e_1 + x_2e_2$ and $y = y_1e_1 + y_2e_2$ belong to V_1 and V_2. We now define a bilinear mapping by

$$\psi(x,y) = x_1y_1f_1 + x_1y_2f_2 + y_2y_1f_3 + x_2y_2f_4 \;.$$

Any vector in U

$$u = \sum_{i=1}^{4} c_if_i$$

is in the image of ψ if and only if it satisfies

$$\text{rank}\begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} < 1 \,.$$

If we pick $u_1 = 2f_1 + 2f_2 + f_3 + f_4$ and $u_2 = f_1 + f_3$, then obviously both of these vectors are in the image of ψ. However, subtracting these two vectors gives

$$u_1 - u_2 = f_1 + 2f_2 + f_4$$

ORIGINAL PAGE IS
OF POOR QUALITY

for which

$$\text{rank}\begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} = 1$$

so that the image is not closed under addition. Since the closure law has been violated, we can conclude that $\text{Im}\ \psi$ does not form a linear subspace of U. The smallest subspace of U which contains the image of ψ is called the subspace generated by the image of ψ and is denoted by $\langle \text{Im}\ \psi \rangle$. This subspace is shown by the dotted contour in Figure 2.1, which depicts the situation for the bilinear case. This minimal subspace becomes a space of tensors when the bilinear map ψ is a tensor product.

A particular subset (m=2) of the set of multilinear mappings is the set of bilinear mappings. A function of two variables is said to be <u>bilinear</u> if it is linear with respect to each of the two variables when the other is fixed. An example of a bilinear function is given by

$$f(x,y) = 3xy\ .$$

This function is linear in each of x and y when the other is fixed as can be readily shown:

$$\begin{aligned} f(x,\alpha y_1 + \beta y_2) &= 3x(\alpha y_1 + \beta y_2) \\ &= \alpha 3xy_1 + \beta 3xy_2 \\ &= \alpha f(x,y_1) + \beta f(x,y_2)\ ; \end{aligned}$$

and

$$\begin{aligned} f(\alpha x_1 + \beta x_2,y) &= 3(\alpha x_1 + \beta x_2)y \\ &= \alpha 3x_1y + \beta 3x_2y \\ &= \alpha f(x_1,y) + \beta f(x_2,y)\ . \end{aligned}$$

We may note that this satisfies the notion that a bilinear mapping is a multilinear mapping with m=2 if both

$$\psi(\alpha x_1 + \beta y_1, y_2) = \alpha\psi(x_1, y_2) + \beta\psi(y_1, y_2) \tag{2.2}$$

and

$$\psi(x_1, \alpha x_2 + \beta y_2) = \alpha\psi(x_1, x_2) + \beta\psi(x_1, y_2) \tag{2.3}$$

hold for all x_i, y_i, α, β as noted in the definition of a general multilinear mapping. A bilinear map is sometimes called 2-linear.

For a given set of vector spaces $V_1, \ldots, V_m$, and U, all multilinear mappings from $V_1 \times V_2 \times \ldots \times V_m$ to U constitute the set $M(V_1, \ldots, V_m : U)$. The set $M(V_1, \ldots, V_m : U)$ is a vector space with addition defined by

$$(\phi + \theta)(v_1, \ldots, v_m) = \phi(v_1, \ldots, v_m) + \theta(v_1, \ldots, v_m) \ ,$$

where the addition on the left side is in $M(V_1, \ldots, V_m : U)$ and addition on the right side is in U. Scalar multiplication is defined by

$$(\alpha\phi)(v_1, \ldots, v_m) = \alpha(\phi(v_1, \ldots, v_m)) \ ,$$

where again $(\alpha\phi)$ represents scalar multiplication in $M(V_1, \ldots, V_m : U)$ and $\alpha(\phi(v_1, \ldots, v_m))$ is scalar multiplication in U. Both of these principles are shown formally in [11]. Also, it is shown that

$$\dim M(V_1, \ldots, V_m : U) = n \prod_{i=1}^{m} n_i \ ,$$

where

$$\dim U = n \text{ and } \dim V_i = n_i \ .$$

Having defined the notion of exactly what comprises a multilinear mapping, we are now in a position to consider the idea of a tensor pro-

ORIGINAL PAGE IS
OF POOR QUALITY

duct. The basic purpose of a tensor product is to convert multilinear mappings (in particular, bilinear mappings) into linear mappings. Basically, this is done because multilinear functions are very complicated objects and are intrinsically more difficult to handle than linear mappings. We now proceed in defining the notion of a tensor product. We consider an arbitrary multilinear mapping

$$\psi : V_1 \times V_2 \times \ldots \times V_m \to U$$

which belongs to $M(V_1,\ldots,V_m:U)$. It can be shown that there exists another multilinear mapping $\nu \in M(V_1,\ldots,V_m:P)$, essentially unique, such that there exists a linear mapping $\mu : P \to U$, which provides

$$\psi = \mu \circ \nu .$$

The tensor product is said to be composed of P and this multilinear mapping ν. We now present the formal definition of the tensor product [10].

A pair (P,ν) is a tensor product of the vector spaces $V_1,\ldots,V_m$ if the following two conditions are satisfied:

(1) $\nu \in M(V_1,\ldots,V_m:P)$ and $\langle \operatorname{Im} \nu \rangle = P$;

(2) if U is any vector space over R, and $\psi \in M(V_1,\ldots,V_m:U)$ is arbitrary, then there exists a $\mu \in L(P:U)$ (that is, it is a linear map from vector space P to vector space U) such that $\psi = \mu \circ \nu$.

The property (1) means that all of the vectors generated by ν form a subspace of P which is equal to P. Property (2) is called the <u>universal factorization property</u>, and is expressed by the commutative diagram in Figure 2.2. It can be shown that the properties (1) and (2)

ORIGINAL PAGE IS
OF POOR QUALITY

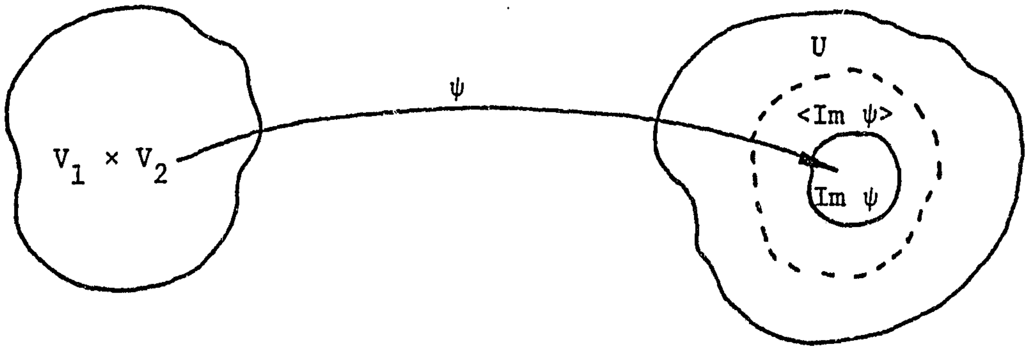

Figure 2.1 The Bilinear Mapping ψ

$$\begin{tikzcd} V_1 \times V_2 \times \cdots \times V_m \arrow[r, "\nu"] \arrow[dr, "\psi"'] & P \arrow[d, "\mu"] \\ & U \end{tikzcd}$$

Figure 2.2 The Tensor Product Idea

ORIGINAL PAGE IS
OF POOR QUALITY

above are equivalent to the following single condition:

(3) for each multilinear map $\psi \in M(V_1,\ldots,V_m:U)$ there is a <u>unique</u> linear map $\mu \in L(P:U)$ such that

$$\psi = \mu \circ \nu .$$

By the way of shorthand notation, we will define the tensor product $\otimes$ as the multilinear mapping ν such that

$$\nu(v_1,v_2,\ldots,v_m) = v_1 \otimes v_2 \otimes \ldots \otimes v_m . \tag{2.4}$$

Therefore, it follows that the space P may be expressed by $V_1 \otimes V_2 \otimes \ldots \otimes V_m$. Rewriting Figure 2.2, we obtain the usual form of the commutative diagram shown in Figure 2.3. It is shown in [11] that for arbitrary vector spaces $V_1,\ldots,V_m$ a tensor product (P,ν) always exists. Also, it can be shown that the tensor product is unique up to an isomorphism.

We now wish to consider the properties of the space $V_1 \otimes V_2 \otimes \ldots \otimes V_m$. Namely, the basis and dimension of this space will be examined. As was mentioned previously, the bilinear case is nothing more than a special case of the multilinear case; so we will first consider constructing a set of basis vectors for the space $V_1 \otimes V_2$. We assume that V_1 and V_2 are both spaces of finite dimension, and that $\dim V_1 = n$ and $\dim V_2 = p$. We also assume that $\{e_1,e_2,\ldots,e_n\}$ is a set of basis vectors for V_1 and that $\{f_1,f_2,\ldots,f_p\}$ is a set of basis vectors for V_2. It is shown in [1] that the tensor products

$$e_i \otimes f_j \quad , \quad \text{where } i = 1,2,\ldots,n \text{ and } j = 1,2,\ldots,p , \tag{2.5}$$

form a system of linearly independent vectors in the space $V_1 \otimes V_2$,

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{tikzcd}
V_1 \times V_2 \times \cdots \times V_m \arrow[r, "\otimes"] \arrow[dr, "\psi"'] & V_1 \otimes V_2 \otimes \cdots \otimes V_m \arrow[d, "\mu"] \\
 & U
\end{tikzcd}
$$

Figure 2.3 Introduction of the Tensor Product Symbol

ORIGINAL PAGE IS
OF POOR QUALITY

which has dimension $n \cdot p$, and therefore constitute a basis for the space. In this proof, it is noted that

$$\dim (V_1 \otimes V_2) = \dim V_1 \cdot \dim V_2 . \tag{2.6}$$

If these results are extended to an arbitrary multilinear mapping, then we may construct a set of basis vectors as follows. Given that $\dim V_k = n_k$, and that $\{e_{ki_k}\}$, $i_k = 1,2,\ldots,n_k$ is a basis for V_k, $k = 1,2,\ldots, m$, then the space $V_1 \otimes V_2 \otimes \ldots \otimes V_m$ has a basis which consists of $e_{1i_1} \otimes e_{2i_2} \otimes \ldots \otimes e_{ki_m}$ and

$$\dim (V_1 \otimes V_2 \otimes \ldots \otimes V_m) = \prod_{i=1}^{m} \dim V_i . \tag{2.7}$$

An example will no doubt clarify the use of multiple indices in this case. Consider the case where $m=3$, $n_1 = \dim V_1 = 2$, $n_2 = \dim V_2 = 2$, and $n_3 = \dim V_3 = 3$. The dimension of $V_1 \otimes V_2 \otimes V_3$ is $2 \cdot 2 \cdot 3 = 12$, so there are accordingly 12 basis vectors, which may be listed as follows:

$$\{e_{11} \otimes e_{21} \otimes e_{31}, e_{11} \otimes e_{21} \otimes e_{32}, e_{11} \otimes e_{21} \otimes e_{33}, e_{11} \otimes e_{22} \otimes e_{31},$$
$$e_{11} \otimes e_{22} \otimes e_{32}, e_{11} \otimes e_{22} \otimes e_{33}, e_{12} \otimes e_{21} \otimes e_{31}, e_{12} \otimes e_{21} \otimes e_{32},$$
$$e_{12} \otimes e_{21} \otimes e_{33}, e_{12} \otimes e_{22} \otimes e_{31}, e_{12} \otimes e_{22} \otimes e_{32}, e_{12} \otimes e_{22} \otimes e_{33}\} .$$

For future results and numerical analyses, we shall place much emphasis on the exact ordering of the basis vectors that can be said to describe a space. In particular, we shall assume that the basis vectors are to be ordered <u>lexicographically</u>. In order to present formally this ordering, we will introduce a few elementary concepts from permutation group theory [12]. First, let $i_1,\ldots,i_m$ be a set of positive integers that satisfy conditions

$$1 \le i_1 \le n;\ 1 \le i_2 \le n;\ \ldots.\ ;\ 1 \le i_m \le n . \tag{2.8}$$

ORIGINAL PAGE IS
OF POOR QUALITY

We shall denote all sequences of these integers by G_n^m. If an integer i belongs to the set G_n^m, this integer has m digits, each of which belongs to a (n+1)-ary number system excluding zero. If we define $|G_n^m|$ as the number of elements in the set G_n^m, clearly $|G_n^m| = n^m$. The following example is offered to illustrate this concept. Arbitarily, we choose $m = 3$ and $n = 2$. According to our definition, $|G_2^3| = 2^3 = 8$, and the elements may be listed as

$$G_2^3 = \{111, 112, 121, 122, 211, 212, 221, 222\} .$$

This illustrates the concept that the set G_n^m actually consists of sequences of integers, m integers in each sequence. The range of each digit in the sequence is from 1 to n. It can be said that the set G_n^m is ordered lexicographically in the example above, that is, if the elements are considered to be an m digit integer, the sequence of elements should start with the smallest number of the base 1) system and strictly monotonically increase to the largest number s is the convention that we will adopt to order our basis vectors. As a final example on the calculation and ordering of basis vectors, we extend these ideas to multiple tensor products of X and U, which are spaces of states and controls. We desire the basis for the space

$$X \otimes X \otimes U$$

where $\dim X = 3$ with basis $\{x_1,x_2,x_3\}$ and $\dim U = 2$ with basis $\{u_1,u_2\}$. The basis is

$$\{x_1 \otimes x_1 \otimes u_1,\ x_1 \otimes x_1 \otimes u_2,\ x_1 \otimes x_2 \otimes u_1,\ x_1 \otimes x_2 \otimes u_2,\ x_1 \otimes x_3 \otimes u_1,$$
$$x_1 \otimes x_3 \otimes u_2,\ x_2 \otimes x_1 \otimes u_1,\ x_2 \otimes x_1 \otimes u_2,\ x_2 \otimes x_2 \otimes u_1,\ x_2 \otimes x_2 \otimes u_2,$$
$$x_2 \otimes x_3 \otimes u_1,\ x_2 \otimes x_3 \otimes u_2,\ x_3 \otimes x_1 \otimes u_1,\ x_3 \otimes x_1 \otimes u_2,\ x_3 \otimes x_2 \otimes u_1,$$

ORIGINAL PAGE IS
OF POOR QUALITY

$x_3 \otimes x_2 \otimes u_2,\ x_3 \otimes x_3 \otimes u_1,\ x_3 \otimes x_3 \otimes u_2\}.$

This is a total of 18 basis vectors. Note that they are listed lexicographically. In general, given the repetition p for X and q for U, the dimension of

$$\underbrace{X \otimes \ldots \otimes X}_{p} \otimes \underbrace{U \otimes \ldots \otimes U}_{q}$$

is

$$|G_n^p| \cdot |G_m^q| \ , \qquad (2.9)$$

where $\dim X = n$ and $\dim U = m$. In this example, note that

$$p = 2 \ ,$$
$$q = 1 \ ,$$
$$n = 3 \ ,$$
$$m = 2 \ ;$$

so,

$$\dim (X \otimes X \otimes U) = |G_3^2| \cdot |G_2^1| = (3^2)(2) = 18 \ .$$

This is obviously just an extension of the case examined in (2.7), but introduces the concept of two different vector spaces which will appear in later problem formulations, since we will always be concerned with spaces of states and controls.

2.2 Tensor Product of Linear Mappings [10]

In this section, we wish to examine the tensor product of two linear mappings. These linear mappings will be defined as follows:

$$A : V_1 \to U_1 \ ,$$
$$B : V_2 \to U_2 \ .$$

ORIGINAL PAGE IS
OF POOR QUALITY

Then, a bilinear map $\psi : V_1 \times V_2 \to U_1 \otimes U_2$ can be defined with the action

$$\psi(v_1,v_2) = (Av_1) \otimes (Bv_2)$$

for $v_1 \in V_1$, $v_2 \in V_2$. It is relatively easy to verify that ψ is bilinear. If we recall the basic definition of bilinearity given in Section 2.1, for $\alpha,\beta \in R$; $w,x \in V_1$; and $y,z \in V_2$,

$$\begin{aligned}\psi(v_1, \alpha y + \beta z) &= (Av_1) \otimes (B(\alpha y + \beta z)) \\ &= (Av_1) \otimes (B\alpha y + B\beta z) \\ &= (Av_1) \otimes (\alpha By + \beta Bz) \ .\end{aligned}$$

Next, we use the fact that the tensor product itself is a bilinear mapping, so as to obtain

$$\begin{aligned}\psi(v_1, \alpha y + \beta z) &= (Av_1) \otimes (\alpha By) + (Av_1) \otimes (\beta Bz) \\ &= \alpha(Av_1) \otimes (By) + \beta(Av_1) \otimes (Bz) \\ &= \alpha\psi(v_1,y) + \beta\psi(v_1,z) \ .\end{aligned}$$

Also, the other half of the proof can be similarly shown, as follows:

$$\begin{aligned}\psi(\alpha w + \beta x, v_2) &= (A(\alpha w + \beta x)) \otimes (Bv_2) \\ &= (\alpha Aw + \beta Ax) \otimes (Bv_2) \\ &= \alpha(Aw) \otimes (Bv_2) + \beta(Ax) \otimes (Bv_2) \\ &= \alpha\psi(w,v_2) + \beta\psi(x,v_2) \ .\end{aligned}$$

We may express these relationships in the commutative diagram shown in Figure 2.4. It is shown in [1], using the contraction property of tensors, that the mapping λ is indeed a linear mapping, and is equal to the tensor product of the two linear mappings A and B. That is,

$$\lambda = A \otimes B \ ,$$

where A and B are as defined previously. This tensor product $A \otimes B$

ORIGINAL PAGE IS
OF POOR QUALITY

is a unique linear map because of the property (3) of the tensor product, which notes the existence of a unique linear map fo each bilinear map, given the commutative diagram structure of Figure 2.4. An example will clarify these relationships. We shall arbitrarily define the linear maps A and B by actions on their basis vectors, and represent these linear maps in the usual matrix form. We assume the following sets of basis vectors exist for V_1, U_1, V_2, and U_2:

$$\text{for } V_1 \;,\; \{e_1, e_2\} \;;$$
$$\text{for } U_1 \;,\; \{f_1, f_2\} \;;$$
$$\text{for } V_2 \;,\; \{g_1, g_2\} \;;$$
$$\text{for } U_2 \;,\; \{h_1, h_2\} \;.$$

We next define

$$A\, e_1 = -\, f_1 + 3f_2 \;,$$
$$A\, e_2 = 2f_1 + f_2 \;,$$

so that

$$[A] = \begin{bmatrix} -1 & 2 \\ 3 & 1 \end{bmatrix} .$$

Next, we define

$$B\, g_1 = h_1$$
$$B\, g_2 = h_1 - h_2$$

so that

$$[B] = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} .$$

We now consider basis vectors for $V_1 \otimes V_2$ and $U_1 \otimes U_2$. Given the above set of basis vectors for V_1, U_1, V_2, and U_2, we may use

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{tikzcd}
V_1 \times V_2 \arrow[r, "\otimes"] \arrow[dr, "\psi"'] & V_1 \otimes V_2 \arrow[d, "\lambda"] \\
 & U_1 \otimes U_2
\end{tikzcd}
$$

Figure 2.4 The Tensor Product of Two Linear Mappings

ORIGINAL PAGE IS
OF POOR QUALITY

the previous ideas on bases for the tensor product to construct the following bases: for $V_1 \otimes V_2$,

$$\{e_1 \otimes g_1, e_1 \otimes g_2, e_2 \otimes g_1, e_2 \otimes g_2\} ;$$

and for $U_1 \otimes U_2$,

$$\{f_1 \otimes h_1, f_1 \otimes h_2, f_2 \otimes h_1, f_2 \otimes h_2\} .$$

The reader will note that the ordering convention that was previously adopted for the basis elements (that is, lexicographic ordering) is employed here. We can next construct $\lambda = A \otimes B$ by looking at the action on the domain basis elements.

$$\begin{aligned}\lambda(e_1 \otimes g_1) &= (A \otimes B)(e_1 \otimes g_1) = (Ae_1) \otimes (Bg_1) \\ &= (-f_1 + 3f_2) \otimes (h_1) \\ &= (-f_1 \otimes h_1) + (3f_2 \otimes h_1) ,\end{aligned}$$

$$\begin{aligned}\lambda(e_1 \otimes g_2) &= (A \otimes B)(e_1 \otimes g_2) = (Ae_1) \otimes (Bg_2) \\ &= (-f_1 + 3f_2) \otimes (h_1 - h_2) \\ &= (-f_1 \otimes h_1) + (f_1 \otimes h_2) + (3f_2 \otimes h_1) - (3f_2 \otimes h_2) ,\end{aligned}$$

$$\begin{aligned}\lambda(e_2 \otimes g_1) &= (A \otimes B)(e_2 \otimes g_1) = (Ae_2) \otimes (Bg_1) \\ &= (2f_1 + f_2) \otimes (h_1) \\ &= (2f_1 \otimes h_1) + (f_2 \otimes h_1) ,\end{aligned}$$

$$\begin{aligned}\lambda(e_2 \otimes g_2) &= (A \otimes B)(e_2 \otimes g_2) = (Ae_2) \otimes (Bg_2) \\ &= (2f_1 + f_2) \otimes (h_1 - h_2) \\ &= (2f_1 \otimes h_1) - (2f_1 \otimes h_2) + (f_2 \otimes h_1) - (f_2 \otimes h_2) .\end{aligned}$$

This implies that the linear mapping $\lambda = A \otimes B$ may be represented by the matrix

ORIGINAL PAGE IS
OF POOR QUALITY

$$[\lambda] = \begin{bmatrix} -1 & -1 & 2 & 2 \\ 0 & 1 & 0 & -2 \\ 3 & 3 & 1 & 1 \\ 0 & -3 & 0 & -1 \end{bmatrix}.$$

This matrix can also be obtained by the following convention.

$$\begin{bmatrix} (\text{Matrix of A})_{11}\ (\text{Matrix of B}) & (\text{Matrix of A})_{12}\ (\text{Matrix of B}) \\ (\text{Matrix of A})_{21}\ (\text{Matrix of B}) & (\text{Matrix of A})_{22}\ (\text{Matrix of B}) \end{bmatrix}$$

This is usually called the <u>Kronecker product</u> of two matrices [13]. With the above convention, it is possible to verify the result for $A \otimes B$, as follows.

$$A \otimes B = \begin{bmatrix} -1 \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} & 2 \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} \\ 3 \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} & 1 \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} -1 & -1 & 2 & 2 \\ 0 & 1 & 0 & -2 \\ 3 & 3 & 1 & 1 \\ 0 & -3 & 0 & -1 \end{bmatrix}.$$

The Kronecker product will be of much use when deriving the expressions for the optimal controller in Section IV. In general, it is not necessary that square matrices be used in computing this product. Given (pxq) and (rxs) matrices, the Kronecker product of these two matrices is defined to be a (pr x qs) matrix. This can be easily seen as a generalization of the above 2x2 case. We also note additional properties of the Kronecker product [14].

(1) The Kronecker product is associative, that is, $(A \otimes B) \otimes C = A \otimes$

ORIGINAL PAGE IS
OF POOR QUALITY

$(B \otimes C)$, where A, B, and C are matrices of not necessarily equal size.

(2) $(AB) \otimes (CD) = (A \otimes C)(B \otimes D)$. This is the factorization property of the Kronecker product.

(3) $A \otimes (B + C) = (A \otimes B) + (A \otimes C)$. The Kronecker product distributes over addition of equal size matrices.

All of the above properties can be easily verified by using generalized matrices and simplifying the result. Note, however, it is not generally true that $A \otimes B = B \otimes A$.

A special case of the Kronecker product occurs when one of the matrices in the Kronecker product is equal to the identity matrix, denoted by 1_k, where k specifies the size of the identity matrix. If we assume that A is $n \times n$ and B is $m \times m$, then we may define the Kronecker sum, $A \oplus B$, as [13]:

$$A \oplus B = A \otimes 1_m + 1_n \otimes B \ . \tag{2.10}$$

As an example, we consider

$$A = \begin{bmatrix} 3 & 4 \\ -1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 2 & -6 \\ 0 & 1 & 1 \\ 3 & -4 & 0 \end{bmatrix}.$$

Then,

$$A \otimes 1_m = \begin{bmatrix} 3 & 4 \\ -1 & 2 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 0 & 4 & 0 & 0 \\ 0 & 3 & 0 & 0 & 4 & 0 \\ 0 & 0 & 3 & 0 & 0 & 4 \\ -1 & 0 & 0 & 2 & 0 & 0 \\ 0 & -1 & 0 & 0 & 2 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 \end{bmatrix};$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$1_n \otimes B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 2 & -6 \\ 0 & 1 & 1 \\ 3 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -6 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 3 & -4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & -6 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 3 & -4 & 0 \end{bmatrix} .$$

Therefore,

$$A \oplus B = \begin{bmatrix} 4 & 2 & -6 & 4 & 0 & 0 \\ 0 & 4 & 1 & 0 & 4 & 0 \\ 3 & -4 & 3 & 0 & 0 & 4 \\ -1 & 0 & 0 & 3 & 2 & -6 \\ 0 & -1 & 0 & 0 & 3 & 1 \\ 0 & 0 & -1 & 3 & -4 & 2 \end{bmatrix} .$$

A special case of the Kronecker summation in (2.10) occurs when $A = B$, that is,

$$A \oplus A = A \otimes 1_n + 1_n \otimes A .$$

We shall find applications of the Kronecker summation idea when we consider the derivation of the necessary controller expressions in the next section. In order to illustrate partially how the Kronecker summation is relevant to system theory problems, we consider the following situation [10]. Let us assume a linear dynamical system

$$\dot{x} = Ax$$

where $A : X \to X$ is a linear map and $x \in X$, where X is a vector space of finite dimension. Recalling the universal factorization property of the tensor product, which allows us to express a bilinear mapping in terms of the tensor product and a unique linear mapping, we have the situation as depicted in Figure 2.5. The previous results have shown that we can consider the linear mapping to analyze the system, since it is known to be unique. Since we have already observed that the tensor product is itself a bilinear function,

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{aligned}
(x \dot{\otimes} x) &= \dot{x} \otimes x + x \otimes \dot{x} \\
&= (Ax) \otimes x + x \otimes (Ax) \\
&= (A \otimes 1_x)\, x \otimes x + (1_x \otimes A)\, x \otimes x \\
&= [(A \otimes 1_x) + (1_x \otimes A)]\, x \otimes x \\
&= (A \oplus A)\, x \otimes x\ .
\end{aligned}
$$

Here, the equality $x = 1_x x$ was employed, where 1_x is the identity mapping of the same dimension as X. One should note the significance of the Kronecker sum term, $A \oplus A$. We shall extend this concept as was applied to a linear system to a nonlinear system in terms of the state variable x and control function u. Since the basic principles involve series expansions, we will expect to require higher order terms in the summation. The simple example presented above will then be seen as a special case of the complete system description which allows for nonlinearities.

2.3 Symmetric Tensor Product

In this section, we present a brief look at the symmetric tensor algebra. In order to understand fully this concept in terms of vector spaces, it is necessary to first present a few of the ideas concerning quotient spaces [10]. Basically, the quotient idea allows us to separate a set into two parts: that which is of interest, and that which is not. In order to pursue this concept further, we recall the principal ideas of an equivalence relation on some set S. Let E be a binary relation on S, that is, E is a subset of $S \times S$. We will denote E by the symbol $\equiv$, which is an equivalence relation on S if the following properties hold.

(i) reflexive, i.e., $s \equiv s$ for all $s \in S$,

(ii) symmetric, i.e., $s_1 \equiv s_2$ implies $s_2 \equiv s_1$, for all s_1,

ORIGINAL PAGE IS
OF POOR QUALITY

$s_2 \in S$,

(iii) transitive, i.e., if $s_1 \equiv s_2$ and $s_2 \equiv s_3$ then this implies $s_1 \equiv s_3$, for $s_1, s_2, s_3 \in S$.

An equivalence relation $\equiv$ divides S into a set of equivalence classes, $S/(\equiv)$ with each class containing elements that are equivalent to each other. Information to be discarded is that which would otherwise distinguish elements in an equivalence class. If we assume that E is an equivalence relation on S, then S/E (read "S modulo E") is the set of equivalence classes. We define the projection operator, π as

$$\pi : S \to S/E$$

with action

$$\pi(s) = s'$$

where $s \in S$ and s' is an equivalence class in S/E to which s is assigned. If we let $f : S \to T$ be a function with the property

$$s_1 \; E \; s_2 \text{ implies } f(s_1) = f(s_2) ,$$

then there is a unique function $g : S/E \to T$, where T is a set, such that

$$g \circ \pi = f .$$

These relationships may be expressed via the following commutative diagram, which is shown in Figure 2.6. This is the "key triangle" that is presented in [15]. The main idea in this presentation is that g is unique for each f. The existence and uniqueness of this function g is shown in [15]. In the algebraic literature, S/E is referred to as a quotient set.

The next step in this sequence is the extension of these concepts

$$
\begin{tikzcd}
X \times X \arrow[r, "\otimes"] \arrow[dr, "\psi"'] & X \otimes X \arrow[d, "\lambda"] \\
& R
\end{tikzcd}
$$

Figure 2.5 Kronecker Summation Motivation

$$
\begin{tikzcd}
S \arrow[r, "\pi"] \arrow[dr, "f"'] & S/E \arrow[d, "g"] \\
& T
\end{tikzcd}
$$

Figure 2.6 The Quotient Set Concept

ORIGINAL PAGE IS
OF POOR QUALITY

on sets to include vector spaces. Suppose V is an F-vector space, and W is a subspace of V. It is possible to define an equivalence relation as follows: for $v_1, v_2 \in V$ and $w \in W$,

$$v_1 \equiv v_2 \quad \text{if} \quad v_1 = v_2 + w \ .$$

The quotient set of the above discussion is replaced by the quotient space V/W, which is also an F-vector space. If we define X as an F-vector space and define the linear map

$$Q : V \rightarrow X \ ,$$

the projection π now becomes a mapping $P : V \rightarrow V/W$. These relationships may be expressed in the commutative diagram shown in Figure 2.7. The unique linear mapping

$$\bar{Q} : V/W \rightarrow X$$

exists if and only if $W \subset \ker Q$. From Figure 2.7, it is also evident that

$$Q = \bar{Q} \circ P \ .$$

We now can extend these concepts surrounding quotient spaces to the subject of symmetric tensors. Namely, we consider the following situation. We recall that the tensor product space $U \otimes U$ has the following set of basis vectors, given that a set of basis vectors for U is $\{b_1, b_2, \ldots, b_m\}$:

$$\begin{gathered} b_1 \otimes b_1 \ , \\ b_1 \otimes b_2 \ , \\ \vdots \\ b_1 \otimes b_m \ , \end{gathered}$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$b_2 \otimes b_m ,$$
$$\vdots$$
$$b_2 \otimes b_m ,$$
$$\vdots$$
$$b_m \otimes b_m . \qquad \text{(this is a total of } m^2 \text{ basis vectors)} .$$

We next define a linear mapping

$$\pi_s : \underbrace{U \otimes \ldots \otimes U}_{p} \to \underbrace{U \otimes \ldots \otimes U}_{p} . \tag{2.11a}$$

This mapping is commonly referred to as the symmetrizer [16] and is defined by

$$\pi_s = \frac{1}{p!} \sum_{\sigma} \sigma \tag{2.11b}$$

where σ denotes a permutation of variables and the sum is made over all possible permutations, with the result being divided by the number of permutations. Basically, permutations of indices arise from the interchange of position of these indices.

First of all, we let Ω be a finite set of arbitrary elements. We define a permutation on Ω as a one-to-one mapping of Ω_m onto Ω, [12] where Ω_m consists of m positive integers $\{1,2,\ldots,m\}$. We let $\Omega = \{k_1,k_2,\ldots,k_m\}$ be a set of elements k_i and we let the permutation operator σ be defined as

$$\sigma = \begin{bmatrix} 1 & 2 & \ldots\ldots & m \\ \sigma(1) & \sigma(2) & \ldots\ldots & \sigma(m) \end{bmatrix} .$$

ORIGINAL PAGE IS
OF POOR QUALITY

Here, the first row specifies the domain of the permutation operator σ and the second row represents an image of σ. As an example,

$$\sigma = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{bmatrix}$$

is one such permutation of $\Omega = \{1,2,3\}$ onto itself. Quite obviously, there are $m!$ permutations for Ω_m. For the above example, we may list all of the members of the image set as

$$\{123,\ 132,\ 213,\ 231,\ 312,\ 321\} .$$

As expected, there are $3! = 6$ members in the set. Returning to the symmetrizer operator, we examine the case where $\dim U = 2$ and $p = 3$. The symmetrizer mapping is

$$\pi_s : U \otimes U \otimes U \to U \otimes U \otimes U .$$

The action on the basis elements of $U \otimes U \otimes U$ is

$$\pi_s(b_1 \otimes b_1 \otimes b_1) = \frac{1}{6}\,(6(b_1 \otimes b_1 \otimes b_1)) = b_1 \otimes b_1 \otimes b_1 ,$$

$$\pi_s(b_1 \otimes b_1 \otimes b_2) = \pi_s(b_1 \otimes b_2 \otimes b_1) = \pi_s(b_2 \otimes b_1 \otimes b_1) =$$
$$\frac{1}{6}\,(b_1 \otimes b_1 \otimes b_2 + b_1 \otimes b_1 \otimes b_2 + b_1 \otimes b_2 \otimes b_1 + b_1 \otimes b_2 \otimes b_1 +$$
$$b_2 \otimes b_1 \otimes b_1 + b_2 \otimes b_1 \otimes b_1) =$$
$$\frac{1}{3}\,(b_1 \otimes b_1 \otimes b_2 + b_1 \otimes b_2 \otimes b_1 + b_2 \otimes b_1 \otimes b_1) ,$$

$$\pi_s(b_1 \otimes b_2 \otimes b_2) = \pi_s(b_2 \otimes b_1 \otimes b_2) = \pi_s(b_2 \otimes b_2 \otimes b_1) =$$
$$\frac{1}{6}\,(b_1 \otimes b_2 \otimes b_2 + b_1 \otimes b_2 \otimes b_2 + b_2 \otimes b_1 \otimes b_2 + b_2 \otimes b_1 \otimes b_2 +$$
$$b_2 \otimes b_2 \otimes b_1 + b_2 \otimes b_2 \otimes b_1) =$$
$$\frac{1}{3}\,(b_1 \otimes b_2 \otimes b_2 + b_2 \otimes b_1 \otimes b_2 + b_2 \otimes b_2 \otimes b_1) ,$$

$$\pi_s\,(b_2 \otimes b_2 \otimes b_2) = \frac{1}{6}\,(6(b_2 \otimes b_2 \otimes b_2)) = b_2 \otimes b_2 \otimes b_2 .$$

For the purposes of this discussion, we will want to consider the case where $p = 2$. We have

ORIGINAL PAGE IS
OF POOR QUALITY

$$\pi_s\,(b_1 \otimes b_1) = b_1 \otimes b_1 \ ,$$
$$\pi_s(b_1 \otimes b_2) = \frac{1}{2}\,(b_1 \otimes b_2 + b_2 \otimes b_1) \ ,$$
$$\pi_s(b_2 \otimes b_1) = \frac{1}{2}\,(b_1 \otimes b_2 + b_2 \otimes b_1) \ ,$$
$$\pi_s(b_2 \otimes b_2) = b_2 \otimes b_2 \ .$$

Next, we define the projection π as follows:

$$\pi : U \otimes U \to U \otimes U/\ker \pi_s \ .$$

The projection operator has the following action on the basis vectors in $U \otimes U$:

$$\pi(b_1 \otimes b_1) = \tilde{b}_1 \vee \tilde{b}_1 \ ,$$
$$\pi(b_1 \otimes b_2) = \tilde{b}_1 \vee \tilde{b}_2 \ ,$$
$$\pi(b_2 \otimes b_1) = \tilde{b}_1 \vee \tilde{b}_2 \ ,$$
$$\pi(b_2 \otimes b_2) = \tilde{b}_2 \vee \tilde{b}_2 \ ,$$

where

$$\tilde{b}_i = \pi(b_i) \ .$$

The wedge operator $\vee$ used here is the symmetric tensor product, which shall be defined shortly. If we express the relationships between π and π_s in the form of a commutative diagram, we have the situation depicted in Figure 2.8. Because of our earlier results we may conclude that there exists a unique linear map $\beta : U \otimes U/\ker \pi_s \to U \otimes U$ with the property

$$\pi_s = \beta \circ \pi \ .$$

If we still assume that the basis elements of the spaces being considered here are listed in lexicographic order, then we may obtain the matrix representations for π_s and π as

ORIGINAL PAGE IS
OF POOR QUALITY

$$\begin{tikzcd}
V \arrow[r, "P"] \arrow[dr, "Q"'] & V/W \arrow[d, "\bar{Q}"] \\
 & X
\end{tikzcd}$$

Figure 2.7 The Quotient Vector Space Concept

$$\begin{tikzcd}
U \otimes U \arrow[r, "\pi"] \arrow[dr, "\pi_s"'] & U \otimes U/\ker \pi_s \arrow[d, "\beta"] \\
 & U \otimes U
\end{tikzcd}$$

Figure 2.8 Operation of the Symmetrizer

ORIGINAL PAGE IS
OF POOR QUALITY

$$[\pi_s] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$[\pi] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Here we note that the use of the brackets around a linear operator implies its matrix representation. Since we choose the basis elements for the factor space $U \otimes U/\ker \pi_s$ as images of basis elements under π_s, then it is possible to determine the matrix representation for β as

$$[\beta] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

It can be easily verified that

$$[\pi_s] = [\beta][\pi]$$

as required.

A few comments are in order concerning symmetric powers of a vector space [16]. If E is an arbitrary vector space, then a vector space

$$\vee^p E \triangleq \underbrace{E \vee E \vee \ldots \vee E}_{p} \tag{2.12a}$$

together with a symmetric p-linear mapping

ORIGINAL PAGE IS
OF POOR QUALITY

$$\vee : \underbrace{E \times \ldots \times E}_{p} \rightarrow \vee^p E \tag{2.12b}$$

is called a p^{th} symmetric power of E if the following conditions are satisfied:

(1) the vectors $\vee (x_1, \ldots, x_p)$ generate $\vee^p E$;

(2) if ψ is any symmetric p-linear mapping of $E \times \ldots \times E$ into an arbitrary vector space F, then there exists a linear map $f : \vee^p E \rightarrow F$ such that $\psi = f \circ \vee$.

The property (1) means that all of the vectors generated by $\vee$ form a subspace of $\vee^p E$ which is equal to $\vee^p E$. Property (2) is the universal factorization property, and is expressed by the commutative diagram shown in Figure 2.9. One notes the surprising similarities between the definition of the tensor algebra that was presented in Section 2.1 and the symmetric tensor algebra presented here. As was done previously, conditions (1) and (2) can be shown to be equivalent to the following single condition:

(3) if ψ is any symmetric mapping of $E \times \ldots \times E$ into F, then there exists a unique linear mapping $f : \vee^p E \rightarrow F$ such that $\psi = f \circ \vee$.

It can be shown that the factor space $U \otimes U/\ker \pi_s$ in Figure 2.8 is isomorphic to the second symmetric power of U, that is to the space $U \vee U$. Redrawing Figure 2.8 with this change, we have the commutative diagram shown in Figure 2.10.

We can next define the symmetric tensor product of two vectors s and t, with $s \in U$ and $t \in U$ as

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{tikzcd}
E^p \arrow[r, "\vee"] \arrow[dr, "\psi"'] & \vee^p E \arrow[d, "f"] \\
& F
\end{tikzcd}
$$

Figure 2.9 The Symmetric Tensor Product

$$
\begin{tikzcd}
U \otimes U \arrow[r, "\pi"] \arrow[dr, "\pi_s"'] & U \vee U \arrow[d, "\beta"] \\
& U \otimes U
\end{tikzcd}
$$

Figure 2.10 Wedge Product Isomorphism

ORIGINAL PAGE IS
OF POOR QUALITY

$$s \vee t = \pi(s \otimes t) .$$

As an example, we will assume that $\dim U = 2$, since we have previously calculated $[\pi]$ for this case:

$$s \vee t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_1 t_1 \\ s_1 t_2 \\ s_2 t_1 \\ s_2 t_2 \end{bmatrix} = \begin{bmatrix} s_1 t_1 \\ s_1 t_2 + s_2 t_1 \\ s_2 t_2 \end{bmatrix}$$

The reader will note that we have expressed the tensor product via the outer or dyadic product rearranged as a 4-vector. The reasoning behind this step will be explored in the following section. Returning to Figure 2.10, we find that we may replace a tensor product

$$u^p \triangleq \underbrace{u \otimes u \otimes \ldots \otimes u}_{p} \tag{2.13}$$

with the linear operator β acting on the symmetric tensor product, that is,

$$u^p = \beta \tilde{u}^p ,$$

where

$$\tilde{u} = \pi(u)$$

and

$$\tilde{u}^p \triangleq \underbrace{u \vee u \vee \ldots \vee u}_{p} .$$

As was previously mentioned, this reduction to the symmetric tensor algebra will be employed whenever we consider the concept of taking derivatives of the tensor product, a topic which is considered in the following section.

ORIGINAL PAGE IS
OF POOR QUALITY

2.4 Derivatives

In this section, we wish to examine a few of the concepts that will be needed when we derive the results for the various controller terms in Section III. In particular, we will consider the concept of taking a partial derivative of a tensor product, which will be needed to solve the fundamental equation of optimality that we are using - the Hamilton-Jacobi-Bellman equation.

Our approach in this section is as follows. First, we present the general definitions of total and partial derivatives and show how these definitions can be applied to a very simple case. These ideas are then extended to the case in which we are particularly interested, which involves the use of the chain rule for abstract derivatives while considering several copies of the vector spaces X and U. By the way of introduction, we may formally define the total derivative as follows [17]. If we let V and W be normed linear spaces with U open in V, a mapping $f : U \to W$ is differentiable at $p \in U$ if there exists $T \in L(V,W)$ so that for $p + x \in U$, $x \in V$, and for $||\cdot||$ an appropriate norm on V and W,

$$\lim_{x \to 0} \frac{||f(p+x) - f(p) - T(x)||}{||x||} = 0 . \tag{2.14}$$

If such a T exists, then T is unique and called the total derivative of f at p, denoted by

$$Df(p)x = Tx . \tag{2.15}$$

Also, suppose we let $V = V_1 \times \ldots \times V_n$, $x = (x_1,\ldots,x_n) \in V$ and U_i open in V_i and consider $f_i : U_i \to W$ with action $x_i \to f(p_1,\ldots,p_{i-1}, x_i, p_{i+1},\ldots,p_n)$. If f_i is differentiable at $p_i \in U_i$, we call its

ORIGINAL PAGE IS
OF POOR QUALITY

derivative the ith partial derivative of f at p, and denote it by $D_i f(p)$, $p \in U_1 \times \ldots \times U_n$. Since we will be concerned primarily with the problem of taking derivatives of the tensor product, we first examine a simple case and note the general concepts involving derivatives. We define the mapping f as follows

$$f : X \times U \rightarrow X \otimes U$$

with action $f(x,u) = x \otimes u$, where

$$x \in X \text{ and } u \in U \; .$$

Provided that the mapping f is differentiable, the total derivative of f is a linear mapping belonging to $L(X \times U, X \otimes U)$ and is defined as [17]

$$Df(\bar{x},\bar{u})(\Delta x,\Delta u) = \lim_{t\to 0} \frac{1}{t}[f(\bar{x} + t\Delta x, \bar{u} + t\Delta u) - f(\bar{x},\bar{u})] \; , \tag{2.16}$$

where $\Delta u \in U$ and $\Delta x \in X$ are the incremental variables while $\bar{u} \in U$ and $\bar{x} \in X$ are the expansion points. For this particular f mapping as defined above,

$$\begin{aligned} Df(\bar{x},\bar{u})(\Delta x,\Delta u) &= \lim_{t\to 0} \frac{1}{t}[(\bar{x} + t\Delta x) \otimes (\bar{u} + t\Delta u) - (\bar{x} \otimes \bar{u})] \\ &= \lim_{t\to 0} \frac{1}{t}[\bar{x} \otimes \bar{u} + \bar{x} \otimes t\Delta u + t\Delta x \otimes \bar{u} + t\Delta x \otimes t\Delta u - \bar{x} \otimes \bar{u}] \\ &= \bar{x} \otimes \Delta u + \Delta x \otimes \bar{u} \; . \end{aligned} \tag{2.17}$$

Next, we shall be concerned with the idea of partial derivatives. First of all, we can define two partial derivatives for this particular case - a partial derivative with respect to x and also with respect to u. We will denote these by D_x and D_u, respectively.

$$\begin{aligned} D_x f(\bar{x},\bar{u})(\Delta x) &= \lim_{t\to 0} \frac{1}{t}[f(\bar{x} + t\Delta x, \bar{u}) - f(\bar{x},\bar{u})] \\ &= \lim_{t\to 0} \frac{1}{t}[(\bar{x} + t\Delta x) \otimes \bar{u} - \bar{x} \otimes \bar{u}] \end{aligned}$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{aligned}
&= \lim_{t\to 0} \frac{1}{t}[\bar{x} \otimes \bar{u} + t\Delta x \otimes \bar{u} - \bar{x} \otimes \bar{u}] \\
&= \Delta x \otimes \bar{u}
\end{aligned} \tag{2.18}
$$

Similarly, the partial derivative with respect to u is computed as

$$
\begin{aligned}
D_u f(\bar{x},\bar{u})(\Delta x,\Delta u) &= \lim_{t\to 0} \frac{1}{t}[f(\bar{x},\bar{u} + t\Delta u) - f(\bar{x},\bar{u})] \\
&= \lim_{t\to 0} \frac{1}{t}[\bar{x} \otimes (\bar{u} + t\Delta u) - \bar{x} \otimes \bar{u}] \\
&= \lim_{t\to 0} \frac{1}{t}[\bar{x} \otimes \bar{u} + \bar{x} \otimes t\Delta u - \bar{x} \otimes \bar{u}] \\
&= \bar{x} \otimes \Delta u \, .
\end{aligned} \tag{2.19}
$$

If we examine the total and partial derivative expressions, it is possible to observe the following concepts. Immediately, it is obvious that the total derivative is nothing more than the sum of all of the possible partial derivatives. This is shown formally in [18] for the general case. Secondly, we can make the following observations:

$$
Df(\bar{x},\bar{u})(0,\Delta u) = D_u f(\bar{x},\bar{u})(\Delta u) \tag{2.20}
$$

and

$$
Df(\bar{x},\bar{u})(\Delta x,0) = D_x f(\bar{x},\bar{u})(\Delta x) \tag{2.21}
$$

The results that we have presented so far are adequate to study only the simple mapping $f(x,u) = x \otimes u$. We must next consider the concepts necessary in order to take derivatives when there exist multiple copies of the spaces X and U, that is, mappings of the form

$$
f : X \times U \to \underbrace{X \otimes \ldots \otimes X}_{p} \otimes \underbrace{U \otimes \ldots \otimes U}_{q} .
$$

In order to examine the general case as stated above, we will need to consider certain concepts involving the chain rule in abstract differentiation [17]. We assume that U_1 is open in V_1, U_2 is open in V_2, $f : U_1 \to U_2$, and $g : U_2 \to W$ for V_1, V_2, W Banach spaces. Let $p \in U_1$

ORIGINAL PAGE IS
OF POOR QUALITY

be such that f is differentiable at p and g is differentiable at $f(p)$. Then $g \circ f : U_1 \to W$ is differentiable at p and

$$D(g \circ f)(p) = Dg(f(p)) \circ Df(p) . \tag{2.22}$$

We will assume that $h = g \circ f$. This chain rule for abstract derivatives is used in [19] to derive the result that is needed concerning partial derivatives of the tensor product. We will assume that X and U are vector spaces such that $h : X \times U \to (X \otimes X) \otimes U$ with action $h(x,u) = x \otimes x \otimes u$. The mapping h has a derivative at some point $p = (\bar{x},\bar{u}) \in X \times U$, where $\bar{x} \in X$ and $\bar{u} \in U$. Also, we define the mapping $f : X \times U \to (X \otimes X) \times U$ with action $f(x,u) = (x \otimes x,u)$ and the mapping $g : (X \otimes X) \times U \to (X \otimes X) \otimes U$ with action

$$g(x \otimes x,u) = x \otimes x \otimes u .$$

Pictorially, we have the commutative diagram displayed in Figure 2.11. We also assume that f is differentiable at p and g is differentiable at $f(p)$. Then,

$$D_x h(\bar{x},\bar{u})(\Delta x) = \Delta x \otimes \bar{x} \otimes \bar{u} + \bar{x} \otimes \Delta x \otimes \bar{u} ,$$

$$D_u h(\bar{x},\bar{u})(\Delta u) = \bar{x} \otimes \bar{x} \otimes \Delta u .$$

We will especially be interested in the partial derivative with respect to the control input, u. If we next allow the mapping h to assume the following generalized form, we can extend these ideas presented above to include this case. First, we assume that

$$h : X \times U \to \underbrace{X \otimes \dots \otimes X}_{p} \otimes \underbrace{U \otimes \dots \otimes U}_{q} ,$$

with action

ORIGINAL PAGE IS
OF POOR QUALITY

$$
\begin{tikzcd}
X \times U \arrow[rr, "h"] \arrow[d, "f"'] & & X \otimes X \otimes U \\
(X \otimes X) \times U \arrow[urr, "g"'] & &
\end{tikzcd}
$$

Figure 2.11 Chain Rule Concept

ORIGINAL PAGE IS
OF POOR QUALITY

$$h(x,u) = \underbrace{x \otimes \ldots \otimes x}_{p} \otimes \underbrace{u \otimes \ldots \otimes u}_{q} .$$

We can compute the required partial derivative as

$$\begin{aligned} D_u h(\bar{x},\bar{u})(\Delta u) &= \underbrace{\bar{x} \otimes \ldots \otimes \bar{x}}_{p} \otimes (\Delta u \otimes \underbrace{\bar{u} \otimes \ldots \otimes \bar{u}}_{q-1} + \bar{u} \otimes \Delta u \otimes \underbrace{\bar{u} \otimes \ldots \otimes \bar{u}}_{q-2} \\ &\quad + \ldots + \underbrace{\bar{u} \otimes \bar{u} \otimes \ldots \otimes \bar{u}}_{q-1} \otimes \Delta u) \\ &= \bar{x}^p \otimes D(\bar{u}^q)(\Delta u) . \end{aligned} \tag{2.23}$$

This is the result that we will need in Section III to solve the necessary minimization problem. Here, as was previously noted, we have assumed that $\bar{x}$ and $\bar{u}$ are expansion points and Δx and Δu are the necessary incremental variables. In order to simplify the notational aspects, we will drop the "bar" notation when we derive the controller results and simply assume that our point of expansion is (x,u).

Since we have presented some introductory results concerning the ideas surrounding the symmetric tensor product, we now consider the partial derivative operation operating on the symmetric product. We consider the following situation:

$$D_u[L(x^p \otimes u^q)](\Delta u) ,$$

where L is a linear map operating on the tensor product $x^p \otimes u^q$. If we apply the chain rule for abstract differentiation plus the fact that the derivative of a linear map is the linear map itself, the partial derivative operator may interchange with the linear map, L, which provides

ORIGINAL PAGE IS
OF POOR QUALITY

$$L\, D_u[x^p \otimes u^q](\Delta u)$$
$$= L\, x^p \otimes D(u^q)(\Delta u) \quad . \tag{2.24}$$

We next substitute

$$u^q = \beta \tilde{u}^q \ ,$$

where it is recalled that β is a linear mapping and

$$\tilde{u}^q = \underbrace{\tilde{u} \vee \tilde{u} \vee \dots \vee \tilde{u}}_{q} .$$

Equation (2.24) then becomes

$$L[(x^p \otimes D(\beta\tilde{u}^q)(\Delta u)]$$
$$= L[x^p \otimes \beta D(\tilde{u}^q)(\Delta u)]$$
$$= L[x^p \otimes \beta(\Delta\tilde{u} \vee \underbrace{\tilde{u} \vee \dots \vee \tilde{u}}_{q-1} + \tilde{u} \vee \Delta\tilde{u} \vee \underbrace{\tilde{u} \vee \dots \vee \tilde{u}}_{q-2} + \dots$$
$$+ \underbrace{\tilde{u} \vee \dots \vee \tilde{u}}_{q-1} \vee \Delta\tilde{u})] \quad . \tag{2.25}$$

Next, we will illustrate the specific uses of the symmetric tensor algebra structure. It can be shown that all of the terms inside the parentheses in equation (2.25) above are equal. Consider the case where $q = 2$ as an example. Equation (2.25) becomes

$$L[x^p \otimes \beta(\Delta\tilde{u} \vee \tilde{u} + \tilde{u} \vee \Delta\tilde{u})] \quad . \tag{2.26}$$

We have previously shown how to compute the symmetric or wedge product of two vectors. Using this technique with $\dim U = 2$, we obtain

$$\Delta\tilde{u} \vee \tilde{u} = \begin{bmatrix} \Delta\tilde{u}_1\tilde{u}_1 \\ \Delta\tilde{u}_2\tilde{u}_1 + \Delta\tilde{u}_1\tilde{u}_2 \\ \Delta\tilde{u}_2\tilde{u}_2 \end{bmatrix}$$

and

ORIGINAL PAGE IS
OF POOR QUALITY

$$\tilde{u} \vee \Delta\tilde{u} = \begin{bmatrix} \tilde{u}_1 \Delta\tilde{u}_1 \\ \tilde{u}_1 \Delta\tilde{u}_2 + \tilde{u}_2 \Delta\tilde{u}_1 \\ \tilde{u}_2 \Delta\tilde{u}_2 \end{bmatrix} .$$

Clearly,

$$\Delta\tilde{u} \vee \tilde{u} = \tilde{u} \vee \Delta\tilde{u} \; .$$

Therefore, equation (2.25) now becomes

$$qL[x^p \otimes \beta(\tilde{u}^{q-1} \vee \Delta\tilde{u})] \; , \qquad (2.27)$$

as there are q identical terms that are added in (2.25). If we recall that

$$\tilde{u} = \pi(u)$$

and

$$\Delta\tilde{u} = \pi(\Delta u) \; ,$$

then (2.27) becomes

$$qL[x^p \otimes \beta(\underbrace{\pi(u) \vee \pi(u) \vee \ldots \vee \pi(u)}_{q-1} \vee \pi(\Delta u))$$

$$= qL[x^p \otimes \beta\pi(\underbrace{u \otimes u \otimes \ldots \otimes u}_{q-1} \otimes \Delta u)] \qquad (2.28)$$

$$= qL[x^p \otimes \pi_s(\underbrace{u \otimes u \otimes \ldots \otimes u}_{q-1} \otimes \Delta u)] \qquad (2.29)$$

$$= qL[x^p \otimes \pi_s(u^{q-1} \otimes \Delta u)] \; . \qquad (2.30)$$

In obtaining equation (2.29), we note the composition

$$\pi_s = \beta \circ \pi$$

was used, as was shown in Figure 2.10. Also, we used a property of the projection in order to obtain (2.28), namely

$$\pi(a_1) \vee \pi(a_2) \vee \ldots \vee \pi(a_n) = \pi(a_1 \otimes a_2 \otimes \ldots \otimes a_n)$$

ORIGINAL PAGE IS
OF POOR QUALITY

for $a_1, \ldots, a_n \in U$.

Basically, we projected into the symmetric tensor algebra in order to symmetrize the derivative result, which allowed us to equate all of the terms in the summation in equation (2.25). After this operation has been accomplished, we effectively return to the (nonsymmetric) tensor algebra with the Δu term in one fixed position. In the following section when we begin the derivation for the optimal controller terms, we will show that the terms that we wish to differentiate partially are of the form presented in (2.24). We will be interested in showing that certain coefficient terms will go to zero for all values of Δu, which is why these terms must be "factored" out of the expression that is of interest.

2.5 Discussion

In this section, we have presented the mathematical preliminaries that are necessary to comprehend the remainder of this work. Each of the concepts presented so far will be of considerable importance whenever we consider the optimal regulation problem in the following sections. Since we will be presented with a minimization problem, we will be particularly concerned with the procedures of partial differentiation with respect to a vector variable. Not surprisingly, the terms that we will be required to differentiate will be expressed via the tensor product, hence, the reason for presenting the material contained in Section 2.4, which is where many of the ideas in the chapter were brought together.

Thus far, our treatment of the subject matter has been highly theoretical and consistently algebraic in nature, with no reference at all to systems concepts. These concepts are presented in the following section, and will appear very similar to some of the classical system theory with which the reader is undoubtedly familiar. The difference, of course, is the use of the multilinear algebra in order to describe a system, a topic that is presented in the first portion of the following section.

III. A NONLINEAR CONTROL PROBLEM

The principal idea of this section is to demonstrate how the concepts presented in the previous section, on the subject of tensor algebra, can be applied to actual control problems involving nonlinear systems. The underlying theme to most of the results presented here is the series expansion of a function about a given point. We begin the section with our basic assumptions concerning the fundamental system description. Having defined these preliminaries, we present series expansion ideas as pertaining to the eventual solution concepts - that is, in terms of matrices operating on basis vectors. A method of calculating the A_{pq} system matrices via Taylor's series ideas is also presented. The next section focuses on the exact procedures that are necessary to derive the needed results, which are the controller gain matrices. By construction, this controller is optimal in nature, satisfying the Hamilton - Jacobi - Bellman (HJB) equation of optimality. In order to perform the necessary minimization, we shall use the concepts presented in Section II concerning the partial differentiation of the tensor product with respect to the control variable u. The recursive nature of the problem is explored, giving rise to solutions for the controller terms as well as to terms in the optimal value function. The last portion of the section provides a partial verification of the derived results as the Linear-Quadratic or LQ problem is verified for the low order terms solution.

ORIGINAL PAGE IS
OF POOR QUALITY

3.1 System Description [1]

In order to proceed with the system representation in terms of tensor expansions, it is necessary to begin with the basic definition of the m^{th} tensor power of a variable. We define

$$u^p = \underbrace{u \otimes u \otimes \ldots \otimes u}_{p \text{ copies}} . \qquad (3.1)$$

The class of nonlinear systems that we will consider are those which can be described by an ordinary differential equation of the usual form

$$\dot{x}(t) = f(x,u,t) \quad , \quad t \in [t_0,t_1] . \qquad (3.2)$$

Here, we assume that $x(\cdot) \in R^n$ is the vector of states and $u(\cdot) \in R^m$ is the vector of controls. The systems to be considered here may be represented in the following generalized form:

$$\dot{x}(t) = \sum_{p,q} A_{pq}(t)\, x^p(t) \otimes u^q(t) \quad , \quad p+q \geq 1 . \qquad (3.3)$$

The A_{pq} terms are linear maps, defined as

$$A_{pq} : (R^n)^p \otimes (R^m)^q \to R^n , \qquad (3.4)$$

where

$$(R^m)^p \triangleq \underbrace{R^m \otimes \ldots \otimes R^m}_{p \text{ copies}} .$$

We next define the performance index, J, as

$$J = \frac{1}{2} M(x(t_1)) + \frac{1}{2} \int_{t_0}^{t_1} L(x(t),\, u(t),\, t)dt , \qquad (3.5)$$

where $M(x(t_1))$ and $L(x(t), u(t), t)$ are positive convex functionals, and $L(x,u,t)$ is assumed to be continuous with respect to t. Also, if $t_1 \to \infty$, it is required that the system is asymptotically stable in

ORIGINAL PAGE IS
OF POOR QUALITY

a sufficiently small neighborhood of the origin. This is the infinite time regulation problem, which can be considered as a special case of the finite time regulation problem that we will consider. Continuing, we assume that the penalty term

$$M(x(t_1)) = M(x_f) \quad , \quad x(t_1) = x_f \ ,$$

is given by

$$M(x_f) = \sum_k M_k \, x_f^{\,k} \quad , \quad k \geq 2 \ . \tag{3.6}$$

The M_k terms are defined as the linear mappings

$$M_k : (R^n)^k \to R \ . \tag{3.7}$$

We let $L(x(t), u(t), t)$ be given by

$$L(x,u,t) = \sum_{j,k} Q_{jk}(t) \, x^j \otimes u^k \quad , \quad j + k \geq 2 \tag{3.8}$$

where

$$Q_{jk}(t) : (R^n)^j \otimes (R^m)^k \to R \ . \tag{3.9}$$

It is now necessary to define the set of admissible control functions as those control functions which can be represented in the usual form of a power series in x, that is

$$u(x,t) = \sum_j K_j(t) \, x^j \quad , \quad j = 1,2,3,\ldots \tag{3.10}$$

where

$$K_j(t) : (R^n)^j \to R^m \ . \tag{3.11}$$

The $K_j(t)$ terms follow the previous convention of being linear maps.

Given the preliminaries presented in this section so far, we are prepared to state the basic optimal control problem, for which a solution

ORIGINAL PAGE IS
OF POOR QUALITY

will be constructed in the next section. We desire to find a suitable control function $u(t)$ such that for all initial conditions x_0 belonging to some open neighborhood of the origin of R^n, the functional J (performance index) is minimized. If such a control function is denoted by $u^*(t)$ and $u(t)$ is any other control function, we require that

$$J(x_0,u^*) \le J(x_0,u) \ .$$

This is to be satisfied for all x_0 in an open neighborhood of the origin. It is shown in [1] that there exists an open neighborhood of the origin such that for all initial conditions in this open neighborhood there exists a $u(t)$ that can be represented in the form of a power series as expressed in (3.10). Moreover, this solution is unique if the solution to the LQ problem is unique. The starting point in the development of this control function will be the Hamilton - Jacobi - Bellman equation, or HJB equation. It is shown in [20] that the HJB equation is a necessary condition for optimality. Before presenting these results, however, it is first necessary to note a few additional principles surrounding the optimal control problem. We define the so called optimal value function $V(x(t),t)$ as follows:

$$V(x,t) = \min_{u \in \Omega} J(x(t),u(s),t) \ , \quad t_0 \le t \le t_1 \quad , \quad s \in [t,t_1] \ , \tag{3.12}$$

such that

$$\dot{x} = f(x,u,t)$$

is satisfied, with initial condition vector

$$x(t_0) = x_0 \ .$$

This function is called the optimal value function because it is equal to

ORIGINAL PAGE IS
OF POOR QUALITY

the minimum value of the performance index J on the interval $[t_0, t_1]$. Also, the set Ω contains control functions that are expressed as in (3.10), that is,

$$u(t) = \sum_j K_j(t)\, x^j(t) \quad , \quad j = 1,2,3,\ldots$$

It can be shown that the optimal value function $V(x,t)$ solves the following functional equation [20]:

$$\min_{u\in\Omega} \left[\frac{d}{dt} V(x,t) + L(x,u,t)\right] = 0 \ . \qquad (3.13)$$

This is the functional form of the HJB equation. It also can be shown [1] that the optimal value function has the following properties:

1) If $x(t_0) = x_0 = 0$, then $V(x,t)$ is identically zero and $u(t) \equiv 0$;

2) there is some open neighborhood of the origin in which $V(x,t)$ can be represented as a power series in the state variable, x. The first term in this series is the quadratic term, so we have

$$V(x,t) = \sum_k V_k(t)\, x^k \quad , \quad k = 2,3,\ldots \ ; \qquad (3.14)$$

3) the boundary condition $V(x_f, t_1) = M(x_f)$ must be satisfied.

Property 1) listed above is really not surprising if we consider the initial conditions as some perturbation from the origin and the state $x(t)$ as being an error term. If this perturbation were not present, there would be no need for any correction mechanism and the control function would be identically zero. Property 2) is very important because it assumes that the optimal value function $V(x,t)$ is available as a power series in the state variable, x, which is similar to those assumptions made for the control function and also for the performance integrand $L(x,u,t)$.

ORIGINAL PAGE IS
OF POOR QUALITY

The underlying principle surrounding the series expansion of the various functions involves the summation of linear mapping terms operating on their respective argument vectors. For example, we assumed

$$\dot{x} = \sum_{p,q} A_{pq}\, x^p \otimes u^q$$

as the given system description. We need to examine the construction of the argument vectors $x^p \otimes u^q$ which will in turn provide information concerning the sizes of the matrices that represent the linear maps A_{pq}. For the purposes of this example, we assume that $\dim U = \dim X = 2$. As an example argument vector, we consider the tensor product

$$x \otimes u \ ,$$

where

$$u = [u_1 \ \ u_2]^T$$

and

$$x = [x_1 \ \ x_2]^T.$$

In order to form the set of basis vectors for $X \otimes U$, we consider the outer or dyadic product xu^T.

$$xu^T = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} [u_1 \ \ u_2] = \begin{bmatrix} x_1u_1 & x_1u_2 \\ x_2u_1 & x_2u_2 \end{bmatrix}.$$

If this 2×2 array is considered to be a 4-dimensional object (that is, a 4-vector) with its elements ordered lexicographically, then it may be listed as the column vector

$$[x_1u_1 \ \ x_1u_2 \ \ x_2u_1 \ \ x_2u_2]^T \ .$$

Alternatively, we can appeal to the bases discussions of the previous section.

ORIGINAL PAGE IS
OF POOR QUALITY

If similar calculations are made for the remaining sets of basis vectors, we can formulate our system description in terms of matrices and vectors. Again, we are assuming, for the purpose of this illustration, that $\dim U = \dim X = 2$, as well as a lexicographic ordering of the basis vectors.

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \underset{A_{10}}{\begin{bmatrix}\cdot & \cdot\\ \cdot & \cdot\end{bmatrix}} \begin{bmatrix}x_1\\ x_2\end{bmatrix} + \underset{A_{01}}{\begin{bmatrix}\cdot & \cdot\\ \cdot & \cdot\end{bmatrix}} \begin{bmatrix}u_1\\ u_2\end{bmatrix} + \underset{A_{11}}{\begin{bmatrix}\cdot & \cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot & \cdot\end{bmatrix}} \begin{bmatrix}x_1u_1\\ x_1u_2\\ x_2u_1\\ x_2u_2\end{bmatrix} +$$

$$\underset{A_{20}}{\begin{bmatrix}\cdot & \cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot & \cdot\end{bmatrix}} \begin{bmatrix}x_1^2\\ x_1x_2\\ x_2x_1\\ x_2^2\end{bmatrix} + \underset{A_{02}}{\begin{bmatrix}\cdot & \cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot & \cdot\end{bmatrix}} \begin{bmatrix}u_1^2\\ u_1u_2\\ u_2u_1\\ u_2^2\end{bmatrix} + \cdots \tag{3.15}$$

Similarly, we can express the series expansions for $V(x,t)$, $L(x,u,t)$, $u(t)$, and $M(x_f)$ as

$$V(x,t) = \underset{V_2}{[\cdot \;\; \cdot \;\; \cdot \;\; \cdot]} \begin{bmatrix}x_1^2\\ x_1x_2\\ x_2x_1\\ x_2^2\end{bmatrix} + \underset{V_3}{[\cdot \;\; \cdot \;\; \cdot \;\; \cdot \;\; \cdot \;\; \cdot \;\; \cdot \;\; \cdot]} \begin{bmatrix}x_1^3\\ x_1^2x_2\\ x_1x_2x_1\\ \vdots\\ x_2^3\end{bmatrix} + \cdots, \tag{3.16}$$

$$L(x,u,t) = \underset{Q_{20}}{[\cdot \;\; \cdot \;\; \cdot \;\; \cdot]} \begin{bmatrix}x_1^2\\ x_1x_2\\ x_2x_1\\ x_2^2\end{bmatrix} + \underset{Q_{02}}{[\cdot \;\; \cdot \;\; \cdot \;\; \cdot]} \begin{bmatrix}u_1^2\\ u_1u_2\\ u_2u_1\\ u_2^2\end{bmatrix} +$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$\underset{Q_{11}}{[\cdot\ \ \cdot\ \ \cdot\ \ \cdot]}\begin{bmatrix} x_1u_1 \\ x_1u_2 \\ x_2u_1 \\ x_2u_2 \end{bmatrix} + \ldots , \tag{3.17}$$

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \underset{K_1}{\begin{bmatrix} \cdot & \cdot \\ \cdot & \cdot \end{bmatrix}}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underset{K_2}{\begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}}\begin{bmatrix} x_1^2 \\ x_1x_2 \\ x_2x_1 \\ x_2^2 \end{bmatrix} + \ldots , \tag{3.18}$$

and

$$M(x_f) = \underset{M_2}{[\cdot\ \ \cdot\ \ \cdot\ \ \cdot]}\begin{bmatrix} x_{1f}^2 \\ x_{1f}x_{2f} \\ x_{2f}x_{1f} \\ x_{2f}^2 \end{bmatrix} + \underset{M_3}{[\cdot\ \ \cdot\ \ \cdot\ \ \cdot\ \ \cdot\ \ \cdot\ \ \cdot\ \ \cdot]}\begin{bmatrix} x_{1f}^3 \\ x_{1f}^2x_{2f} \\ \cdot \\ \cdot \\ \cdot \\ x_{2f}^3 \end{bmatrix} + \ldots . \tag{3.19}$$

We should emphasize once again that the representations of these linear maps as just illustrated are inherently dependent upon the particular ordering convention that is employed for the basis elements. We will consistently employ the lexicographic ordering convention in all discussions. It also would be useful to relate the matrices that we have defined here to the "usual" matrices found in the classical optimal control problem. In the traditional approach, a quadratic $L(x,u,t)$ would be described as

$$L(x,u,t) = x^TQx + u^TRu + x^TCu . \tag{3.20}$$

In order to show the relationship, say, between the Q matrix above and

ORIGINAL PAGE IS
OF POOR QUALITY

the Q_{20} vector that appeared in (3.17), we assume that dim U = dim X = 2, and that Q is represented by

$$Q = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix}.$$

Then,

$$\begin{aligned} x^T Q x &= [x_1 \; x_2] \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= [q_{11} x_1^2 + q_{12} x_1 x_2 + q_{21} x_2 x_1 + q_{22} x_2^2] \\ &= [q_{11} \; q_{12} \; q_{21} \; q_{22}] \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_2 x_1 \\ x_2^2 \end{bmatrix} \\ &= Q_{20} \, x^2 . \end{aligned}$$

We can similarly determine the components of the other Q_{ij} terms as needed from their corresponding matrices in the classical optimal control problem.

Thus far, we have defined the A_{pq} system matrices, but have not specified how these matrices may be obtained from a given nonlinear system. Basically, there are two methods by which these matrices may be obtained. The first is simply obtaining them by inspection. Unfortunately, this method can only be applied where the variables x and u do not appear as arguments for other functions. The following example illustrates this method. We assume that we are given the system

$$\dot{x}_1 = 3x_1 + 4u_2 + 2x_2u_1 - 5x_1^2 + x_1u_2 ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$\dot{x}_2 = -u_1 + x_1 - 2u_1^2 + 7u_2^2 + u_1u_2 \ .$$

We can calculate the system matrices by inspection.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 3 & 0 \\ 1 & 0 \end{bmatrix}}_{A_{10}} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 4 \\ -1 & 0 \end{bmatrix}}_{A_{01}} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{11}} \begin{bmatrix} x_1u_1 \\ x_1u_2 \\ x_2u_1 \\ x_2u_2 \end{bmatrix} +$$

$$\underbrace{\begin{bmatrix} -5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{20}} \begin{bmatrix} x_1^2 \\ x_1x_2 \\ x_2x_1 \\ x_2^2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 7 \end{bmatrix}}_{A_{02}} \begin{bmatrix} u_1^2 \\ u_1u_2 \\ u_2u_1 \\ u_2^2 \end{bmatrix}$$

For systems where the variables x and u appear as arguments for other functions, we must use the other approach, which involves a Taylor's series expansion of a function in two vector variables. The theory behind this concept is discussed quite extensively in [1] and [16] and involves the traditional Taylor's series approach but developed in terms of vector-valued tensors and the contraction operator. Another approach that defines the series expansion form in terms of tensor products may be found in [9]. The details of these methods are not really relevant to our results here, but the ideas surrounding the definition of the various system matrices are quite important. For the purposes of this example, we note that we are still assuming that $\dim U = \dim X = 2$. The A_{pq} system matrices can be computed via the partial derivative operation, as follows:

$$A_{10} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} \Bigg|_{(x_0,u_0)} , \quad A_{01} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} \end{bmatrix} \Bigg|_{(x_0,u_0)} ,$$

$$A_{11} = \begin{bmatrix} \frac{\partial^2 f_1}{\partial x_1 \partial u_1} & \frac{\partial^2 f_1}{\partial x_1 \partial u_2} & \frac{\partial^2 f_1}{\partial x_2 \partial u_1} & \frac{\partial^2 f_1}{\partial x_2 \partial u_2} \\ \frac{\partial^2 f_2}{\partial x_1 \partial u_1} & \frac{\partial^2 f_2}{\partial x_1 \partial u_2} & \frac{\partial^2 f_2}{\partial x_2 \partial u_1} & \frac{\partial^2 f_2}{\partial x_2 \partial u_2} \end{bmatrix} \Bigg|_{(x_0,u_0)} ,$$

$$A_{20} = \frac{1}{2} \begin{bmatrix} \frac{\partial^2 f_1}{\partial x_1^2} & \frac{\partial^2 f_1}{\partial x_1 \partial x_2} & \frac{\partial^2 f_1}{\partial x_2 \partial x_1} & \frac{\partial^2 f_1}{\partial x_2^2} \\ \frac{\partial^2 f_2}{\partial x_1^2} & \frac{\partial^2 f_2}{\partial x_1 \partial x_2} & \frac{\partial^2 f_2}{\partial x_2 \partial x_1} & \frac{\partial^2 f_2}{\partial x_2^2} \end{bmatrix} \Bigg|_{(x_0,u_0)} ,$$

$$A_{02} = \frac{1}{2} \begin{bmatrix} \frac{\partial^2 f_1}{\partial u_1^2} & \frac{\partial^2 f_1}{\partial u_1 \partial u_2} & \frac{\partial^2 f_1}{\partial u_2 \partial u_1} & \frac{\partial^2 f_1}{\partial u_2^2} \\ \frac{\partial^2 f_2}{\partial u_1^2} & \frac{\partial^2 f_2}{\partial u_1 \partial u_2} & \frac{\partial^2 f_2}{\partial u_2 \partial u_1} & \frac{\partial^2 f_2}{\partial u_2^2} \end{bmatrix} \Bigg|_{(x_0,u_0)} ,$$

$$\cdots\cdot \quad .$$

The other values of the A_{pq} system matrices that are needed (depending on the problem degree) can be calculated similarly. The point (x_0,u_0) is assumed to be the expansion point. An interesting feature regarding the above way of calculating the system matrices is how the sequence of indices increases lexicographically from left to right in each row. This feature is, of course, basis dependent.

One of the concepts surrounding the HJB equation is the idea of minimization. This minimization will take place after appropriate substitutions for $V(x,t)$ and $L(x,u,t)$ are made. Since we are presented with a function of two variables, the control u and state variable x, we will perform the minimization operation by taking the necessary partial derivatives and setting the result to zero. As we shall see, this will involve the concepts of Section 2.4 where we examined the possibility of taking partial derivatives of tensor products of two variables. We also note that the particular type of control functions that we desire are the so called analytic feedback controllers [1]. This requirement is quite important in obtaining the final solution. Toward this end, we begin the derivation of the controller expressions in the next portion of this section.

3.2 Derivation of Controller Expressions

In this section, we show how to derive expressions for the controller as a function of the system description and the performance index terms. The solution will be constructed in a recursive manner by solving the HJB equation for the unknown coefficient matrices

$$V_k(t) \quad , \quad k = 2,3,\ldots$$

and

$$K_j(t) \quad , \quad j = 1,2,3,\ldots$$

such that the boundary condition

$$V(x(t_1),t_1) = M(x(t_1))$$

is met. Beginning with the HJB equation

$$\min_{u\in\Omega} \left[\frac{d}{dt} V(x,t) + L(x,u,t)\right] = 0 \ ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

we first examine how to compute the time derivative of the optimal value function, $V(x,t)$, where we recall from (3.16) that $V(x,t)$ can be expressed in series expansion form as

$$V(x,t) = V_2 x^2 + \dot{V}_3 x^3 + \dots \quad .$$

Therefore,

$$\frac{d}{dt} V(x,t) = \dot{V}_2 x^2 + V_2(\dot{x} \otimes x + x \otimes \dot{x}) + \dot{V}_3 x^3 + V_3(\dot{x} \otimes x \otimes x + x \otimes \dot{x} \otimes x + x \otimes x \otimes \dot{x}) + \dots \tag{3.21}$$

because of the bilinearity of the tensor product and using the ordinary product rule. In general, we note the movement of the $\dot{x}$ then from left to right in the summation and express the general result as

$$\frac{d}{dt} V(x,t) = \sum_{k=2}^{\infty} V_k(t) \sum_{j=1}^{k} \underbrace{x \otimes x \otimes \dots x}_{\substack{j-1 \\ \text{copies}}} \otimes \dot{x} \otimes \underbrace{x \otimes \dots \otimes x}_{\substack{k-j \\ \text{copies}}} + \sum_{k=2}^{\infty} \dot{V}_k(t)\, x^k \; . \tag{3.22}$$

We now examine the substitution for the $\dot{x}$ terms, namely

$$\dot{x} = \sum_{p,q} A_{pq}\, x^p \otimes u^q \quad , \quad p + q \geq 1 \; .$$

Making this substitution, and including the expansion terms for $L(x,u,t)$, we arrive at the expression in the HJB equation that is to be minimized with respect to the vector variable u, namely

$$\frac{d}{dt} V(x,t) + L(x,u,t) =$$

$$\sum_{k=2}^{\infty} V_k(t) \sum_{j=1}^{k} \underbrace{x \otimes \dots \otimes x}_{\substack{j-1 \\ \text{copies}}} \otimes \Big(\sum_{p,q} A_{pq}\, x^p \otimes u^q\Big) \otimes \underbrace{x \otimes \dots \otimes x}_{\substack{k-j \\ \text{copies}}} + \sum_{k=2}^{\infty} \dot{V}_k(t)\, x^k$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$+ \sum_{m,n} Q_{mn}\, x^m \otimes u^n \ , \qquad (3.23)$$

where $m+n \geq 2$ and $p+q \geq 1$. Recalling the various properties of the Kronecker product outlined in Section II we may proceed with the simplification. Equation (3.23) then becomes

$$\sum_{k=2}^{\infty} V_k(t) \sum_{j=1}^{k} \sum_{p,q} (1_x^{j-1} \otimes A_{pq} \otimes 1_x^{k-j})(x^{j-1} \otimes (x^p \otimes u^q) \otimes x^{k-j})$$
$$+ \sum_{k=2}^{\infty} \dot{V}_k(t)\, x^k + \sum_{m,n} Q_{mn}\, x^m \otimes u^n \ , \qquad (3.24)$$

where we define the multiple Kronecker product of the identity matrix 1_x^{k-j} as

$$1_x^{k-j} = \underbrace{1_x \otimes 1_x \otimes \ldots \otimes 1_x}_{k-j \text{ copies}} .$$

It would be very advantageous to be able to alter the ordering of the terms in the above expression and group the x^{j-1}, x^{k-j}, and x^p terms to obtain a term x^{p+k-1}[16]. However, in order to accomplish this, it will be necessary to introduce a new operator. We define the permutation matrix in a slightly modified manner from the definition in [21], but with similar results. Let

$$S_{pq}^{r}\ (x^p \otimes u^q) \triangleq x^{p-r} \otimes u^q \otimes x^r \ , \qquad (3.25)$$

where

$$r \leq p \ .$$

If we make this substitution in equation (3.24), we have

$$\frac{d}{dt} V(x,t) + L(x,u,t) = \sum_{k=2}^{\infty} V_k(t) \sum_{j=1}^{k} \sum_{p,q} (1_x^{j-1} \otimes A_{pq} \otimes 1_x^{k-j}) S_{\ell q}^{k-j}$$
$$(x^{\ell} \otimes u^q) + \sum_{k=2}^{\infty} \dot{V}_k(t)\, x^k + \sum_{m,n} Q_{mn}\, x^m \otimes u^n \ , \qquad (3.26)$$

ORIGINAL PAGE IS
OF POOR QUALITY

where we define the index ℓ as

$$\ell = p + k - 1 .$$

Also, we note that $S^0_{\ell q}$ (r=0) is trivially the identity matrix, of dimension equal to the dimension of

$$\underbrace{X \otimes \ldots \otimes X}_{\ell} \otimes \underbrace{U \otimes \ldots \otimes U}_{q} ,$$

which is equal to

$$n^{\ell} \cdot m^{q}$$

for $\dim X = n$ and $\dim U = m$.

An example will be useful in observing the action of this permutation matrix. For the purposes of this example, we shall assume that

$$k = 2 ,$$

$$p = 1 ,$$

and

$$q = 1 .$$

Then, we have, from (3.26)

$$V_2 \, [(A_{11} \otimes 1_x) \, S^1_{21} \, x^2 \otimes u + (1_x \otimes A_{11}) \, (x^2 \otimes u)]$$

$$= V_2 \, [(A_{11} \otimes 1_x) \, S^1_{21} + (1_x \otimes A_{11})] \, x^2 \otimes u .$$

In general, the S^r_{pq} operators are linear mappings, and are represented in matrix form by a $n^p \cdot m^q$ square matrix, where $\dim X = n$ and $\dim U = m$. These matrices have a single "1" in each column and basically reorder the listing of the elements in a particular basis vector. For example, S^1_{11} can be represented as

ORIGINAL PAGE IS
OF POOR QUALITY

$$[S_{11}^1] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} .$$

For higher order p and q, there is an ambiguity that must be resolved before we can continue. As an example, we consider the operator S_{21}^1, with action

$$S_{21}^1 \ (x^2 \otimes u) = x \otimes u \otimes x \ .$$

In matrix form, we have:

$$\begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix} \begin{bmatrix} x_1x_1u_1 \\ x_1x_1u_2 \\ x_1x_2u_1 \\ x_1x_2u_2 \\ x_2x_1u_1 \\ x_2x_1u_2 \\ x_2x_2u_1 \\ x_2x_2u_2 \end{bmatrix} = \begin{bmatrix} x_1u_1x_1 \\ x_1u_1x_2 \\ x_1u_2x_1 \\ x_1u_2x_2 \\ x_2u_1x_1 \\ x_2u_1x_2 \\ x_2u_2x_1 \\ x_2u_2x_2 \end{bmatrix} .$$

The ambiguity arises in the above example in the construction of rows 2 and 5, as well as rows 4 and 7. We shall adopt the convention that the x vector components will remain in the same order whenever the tensor product is reordered. Thus, in row 2 of the above matrix, we choose the product $x_1x_2u_1$ to equate to $x_1u_1x_2$, rather than the product $x_2x_1u_1$. This convention will be consistently employed throughout the remainder of the work.

Now that we have clarified the necessary issues, we are ready to proceed with the solution. We recall that the first step in solving the optimal regulation problem will be to calculate

ORIGINAL PAGE IS
OF POOR QUALITY

$$D_u \left[\frac{d}{dt} V(x,t) + L(x,u,t)\right] (\Delta u) = 0 \ . \tag{3.27}$$

It is necessary to consider what happens whenever equation (3.26) is partially differentiated with respect to the control variable u. So that we may differentiate term-by-term, we shall first list a few of the initial terms in (3.26):

$$\begin{aligned}
\frac{d}{dt}V(x,t) + L(x,u,t) &= \dot{V}_2 x^2 + V_2\{(A_{10} \oplus A_{10})x^2 + [(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})] \\
&\quad x \otimes u + [(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})]x^2 \otimes u + [(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})] \\
&\quad x \otimes u^2 + [(A_{20} \otimes 1_x) + (1_x \otimes A_{20})]x^3 + \dots\} + \dot{V}_3 x^3 + V_3\{A_{10} \oplus A_{10} \oplus A_{10})x^3 + \\
&\quad [(A_{01} \otimes 1_x{}^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1_x{}^2 \otimes A_{01})]x^2 \otimes u + \\
&\quad [(A_{11} \otimes 1_x{}^2)S^2_{31} + (1_x \otimes A_{11} \otimes 1_x)S^1_{31} + (1_x{}^2 \otimes A_{11})]x^3 \otimes u + \dots\} + \dots \\
&\quad + Q_{02}\, u^2 + Q_{20}\, x^2 + Q_{11}\, x \otimes u + \dots \ .
\end{aligned} \tag{3.28}$$

In listing these terms, we have used the Kronecker summation that was previously defined in Section II that is,

$$A \oplus B \triangleq A \otimes 1_m + 1_\ell \otimes B \ ,$$

where A is $\ell \times \ell$ and B is $m \times m$. This sum is well-defined because $A \otimes 1_m$ is an $\ell m \times \ell m$ matrix as well as $1_\ell \otimes B$. By using this definition, we can write, for C $n \times n$,

$$\begin{aligned}
&A \otimes 1_m \otimes 1_n + 1_\ell \otimes B \otimes 1_n + 1_\ell \otimes 1_m \otimes C \\
&= (A \otimes 1_m + 1_\ell \otimes B) \otimes 1_n + 1_\ell \otimes 1_m \otimes C \\
&= (A \oplus B) \otimes 1_n + (1_\ell \otimes 1_n) \otimes C \\
&= A \oplus B \oplus C \ .
\end{aligned}$$

This is usually called the multiple Kronecker summation [1] and can be viewed as an extension of the normal Kronecker sum.

If we recall the process of partially differentiating linear map-

ORIGINAL PAGE IS
OF POOR QUALITY

pings operating on tensor products that was outlined in Section 2.4, we can differentiate (3.28) term by term:

$$D_u \{V_2[(A_{01} \otimes 1_x) S_{11}^1 + (1_x \otimes A_{01})] x \otimes u\} (\Delta u)$$
$$= [(A_{01} \otimes 1_x) S_{11}^1 + (1_x \otimes A_{01})] x \otimes \Delta u ,$$
$$D_u [Q_{11} x \otimes u] (\Delta u) = Q_{11} x \otimes \Delta u ,$$

and

$$D_u [Q_{02} u^2] (\Delta u)$$
$$= Q_{02} D(u^2)(\Delta u)$$
$$= 2 Q_{02} \pi_s (u \otimes \Delta u) .$$

If similar calculations are carried out for the remaining terms in (3.28), we may list the result as follows:

$$\begin{aligned} D_u [\tfrac{d}{dt} V(x,t) + L(x,u,t)] (\Delta u) = \{&2 Q_{02} \pi_s (K_1 \otimes 1_u) + Q_{11} + \\ V_2 [(A_{01} \otimes 1_x) S_{11}^1 + (1_x \otimes A_{01})]\} x \otimes \Delta u + \{&2 Q_{02} \pi_s (K_2 \otimes 1_u) \\ + 3 Q_{03} \pi_s (K_1 \otimes K_1 \otimes 1_u) + 2 Q_{12} [1_x \otimes \pi_s (K_1 \otimes 1_u)] + Q_{21} + \\ V_2 [(A_{11} \otimes 1_x) S_{21}^1 + (1_x \otimes A_{11})] + 2 V_2 [(A_{02} \otimes 1_x) S_{12}^1 + (1_x \otimes A_{02})] \\ [1_x \otimes \pi_s (K_1 \otimes 1_u)] + V_3 [(A_{01} \otimes 1_x^2) S_{21}^2 + (1_x \otimes A_{01} \otimes 1_x) S_{21}^1 + \\ (1_x^2 \otimes A_{01})]\} x^2 \otimes \Delta u + \ldots = 0 \quad . \end{aligned} \tag{3.29}$$

In order to clarify the origins of the terms listed in equation (3.29), refer to Figures 3.1 and 3.2, which show the original terms of equation (3.28) and their values after differentiation.

The HJB equation thus far has been solved for the optimal control, which can be expressed as a function of x and t. We may write

$$u^*(x,t) = \sum_j K_j^*(t) \; x^j \quad , \quad j = 1,2,3,\ldots \quad , \tag{3.30}$$

where $u^*(x,t)$ is the optimal control. If this substitution is made,

Original term	After partial differentation	First order terms (those that multiply $x \otimes \Delta u$)	Second order terms (those that multiply $x^2 \otimes \Delta u$)
$Q_{02}\ u^2$	$2\ Q_{02}\ \pi_s\ (u \otimes \Delta u)$	$2\ Q_{02}\ \pi_s\ (K_1 \otimes 1_u)$	$2\ Q_{02}\ \pi_s\ (K_2 \otimes 1_u)$
$Q_{20}\ x^2$	....	....	....
$Q_{11}\ x \otimes u$	$Q_{11}\ x \otimes \Delta u$	Q_{11}	....
$Q_{03}\ u^3$	$3\ Q_{03}\ \pi_s\ (u^2 \otimes \Delta u)$	....	$3\ Q_{03}\ \pi_s\ (K_1 \otimes K_1 \otimes 1_u)$
$Q_{12}\ x \otimes u^2$	$2\ Q_{12}\ x \otimes \pi_s\ (u \otimes \Delta u)$	....	$2\ Q_{12}\ [1_x \otimes \pi_s\ (K_1 \otimes 1_u)]$
$Q_{21}\ x^2 \otimes u$	$Q_{21}\ x^2 \otimes \Delta u$	....	Q_{21}
$Q_{30}\ x^3$	....	....	....
$\vdots$	$\vdots$	$\vdots$	$\vdots$

Figure 3.1

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

Original term	After partial differentation	First order terms (those that multiply $x \otimes \Delta u$)	Second order terms (those that multiply $x^2 \otimes \Delta u$)
$\dot{V}_2\ x^2$	....	....	....
$V_2[(A_{01}\otimes 1_x)S^1_{11} + (1_x\otimes A_{01})]x\otimes u$	$V_2[(A_{01}\otimes 1_x)S^1_{11} + (1_x\otimes A_{01})]\ x\otimes\Delta u$	$V_2[(A_{01}\otimes 1_x)S^1_{11}+(1_x\otimes A_{01})]$	....
$V_2(A_{10}\oplus A_{10})x^2$	....	....	....
$V_2[(A_{11}\otimes 1_x)S^1_{21} +(1_x\otimes A_{11})]x^2\otimes u$	$V_2[(A_{11}\otimes 1_x)S^1_{21} + (1_x\otimes A_{11})]\ x^2\otimes\Delta u$	....	$V_2[(A_{11}\otimes 1_x)S^1_{21} + (1_x\otimes A_{11})]$
$V_2[(A_{02}\otimes 1_x)S^1_{12} +(1_x\otimes A_{02})]x\otimes u^2$	$2\ V_2[(A_{02}\otimes 1_x)S^1_{12}+(1_x\otimes A_{02})]\ x\otimes\pi_s(u\otimes\Delta u)$	....	$2\ V_2[(A_{02}\otimes 1_x)S^1_{12}+(1_x\otimes A_{02})]\ [1_x\otimes\pi_s(K_1\otimes 1_u)]$
$V_2[(A_{20}\otimes 1_x) + (1_x\otimes A_{20})]\ x^3$	....	....	....
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\dot{V}_3\ x^3$	....	....	....
$V_3[(A_{01}\otimes 1_x{}^2)S^2_{21}+(1_x\otimes A_{01}\otimes 1_x)S^1_{21}+(1_x{}^2\otimes A_{01})]x^2\otimes u$	$V_3[(A_{01}\otimes 1_x^2)S^2_{21}+(1_x\otimes A_{01}\otimes 1_x)S^1_{21}+(1_x{}^2\otimes A_{01})]x^2\otimes\Delta u$	....	$V_3[(A_{01}\otimes 1_x{}^2)S^2_{21}+(1_x\otimes A_{01}\otimes 1_x)S^1_{21}+(1_x{}^2\otimes A_{01})]$

Figure 3.2

the resulting equation is (3.29). We will suppress the asterisk notation and simply note the optimal gain matrices as $K_j(t)$. A few comments are in order concerning the multiple tensor product of the control vector when the substitution (3.30) is made. We will first consider the product $u \otimes u$.

$$u \otimes u = (\sum_j K_j(t)\, x^j) \otimes (\sum_k K_k(t)\, x^k)$$

$$= \sum_{j,k} (K_j(t) \otimes K_k(t))\, x^{j+k} .$$

Likewise, the general form is therefore

$$u^p = \sum_{j_1, j_2, \ldots, j_p} (K_{j_1}(t) \otimes K_{j_2}(t) \otimes \ldots \otimes K_{j_p}(t))\, x^{j_1+j_2+\ldots+j_p} . \quad (3.31)$$

By using these techniques and substitutions, it is possible to obtain equation (3.29).

The factored form of equation (3.29) presents a possible method of solution. We desire that this entire expression be identically zero for all values of Δu. We can express the tensor products $x \otimes \Delta u$ and $x^2 \otimes \Delta u$ in matrix form as

$$\chi_1\ \Delta u \quad , \quad \chi_2\ \Delta u \ .$$

For the case where $\dim U = \dim X = 2$,

$$\chi_1 = \begin{bmatrix} x_1 & 0 \\ 0 & x_1 \\ x_2 & 0 \\ 0 & x_2 \end{bmatrix} ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$\chi_2 = \begin{bmatrix} x_1^2 & 0 \\ 0 & x_1^2 \\ x_1x_2 & 0 \\ 0 & x_1x_2 \\ x_2x_1 & 0 \\ 0 & x_2x_1 \\ x_2^2 & 0 \\ 0 & x_2^2. \end{bmatrix},$$

and

$$\Delta u = \begin{bmatrix} \Delta u_1 \\ \Delta u_2 \end{bmatrix}.$$

Therefore, for an identically zero result, one of the requirements is that

$$\{2\, Q_{02}\, \pi_s\, (K_1 \otimes 1_u) + Q_{11} + V_2\, [(A_{01} \otimes 1_x)S_{11}^1 + (1_x \otimes A_{01})]\}\, \chi_1 = 0. \quad (3.32)$$

Since the above quantity that multiplies χ_1 is a row vector with four components, the general form is

$$[A_1\ A_2\ A_3\ A_4] \begin{bmatrix} x_1 & 0 \\ 0 & x_1 \\ x_2 & 0 \\ 0 & x_2 \end{bmatrix} = [0\quad 0]\ ,$$

or

$$\begin{bmatrix} A_1 & A_3 \\ A_2 & A_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

which implies that

ORIGINAL PAGE IS
OF POOR QUALITY

$$A_1 = A_2 = A_3 = A_4 = 0 \ ,$$

since A_i is independent of x. This means that our requirement reduces to

$$2\, Q_{02}\, \pi_s\, (K_1 \otimes 1_u) + Q_{11} + V_2\, [(A_{01} \otimes 1_x) S_{11}^1 + (1_x \otimes A_{01})] = 0 \ . \qquad (3.33)$$

It is possible to extend this reasoning to include terms of higher order. We can express the tensor product $x^2 \otimes \Delta u$ in matrix form as

$$\chi_2\, \Delta u \ .$$

This means that we must require

$$\begin{aligned}
\{2\, Q_{02}\, \pi_s\, (K_2 \otimes 1_u) + 3\, Q_{03}\, \pi_s\, (K_1 \otimes K_1 \otimes 1_u) + 2\, Q_{12}\, [1_x \otimes \pi_s\, (K_1 \otimes 1_u)] + \\
Q_{21} + V_2\, [(A_{11} \otimes 1_x) S_{21}^1 + (1_x \otimes A_{11})] + \\
2\, V_2\, [(A_{02} \otimes 1_x) S_{12}^1 + (1_x \otimes A_{02})]\, [1_x \otimes \pi_s\, (K_1 \otimes 1_u)] + \\
V_3\, [(A_{01} \otimes 1_x^{\,2}) S_{21}^2 + (1_x \otimes A_{01} \otimes 1_x) S_{21}^1 + (1_x^{\,2} \otimes A_{01})]\, \chi_2 = 0 \ . \qquad (3.34)
\end{aligned}$$

The above quantity that multiplies χ_2 is a row vector with eight components, which can be represented as

$$[B_1\ B_2\ B_3\ B_4\ B_5\ B_6\ B_7\ B_8] \begin{bmatrix} x_1^{\,2} & 0 \\ 0 & x_1^{\,2} \\ x_1 x_2 & 0 \\ 0 & x_1 x_2 \\ x_2 x_1 & 0 \\ 0 & x_2 x_1 \\ x_2^{\,2} & 0 \\ 0 & x_2^{\,2} \end{bmatrix} = [0 \quad 0] \ .$$

Equivalently, we may write

ORIGINAL PAGE IS
OF POOR QUALITY

$$\begin{bmatrix} B_1 & B_3 & B_5 & B_7 \\ B_2 & B_4 & B_6 & B_8 \end{bmatrix} \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_2 x_1 \\ x_2^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ,$$

which implies a sufficient condition of

$$B_i = 0 \quad , \quad \text{for} \quad i = 1,2,\ldots,8 \quad .$$

In order to show that the above condition is also necessary, we would need to answer questions regarding symmetry and the image of a bilinear function. Although the B_i components above are independent of x, we cannot argue that it is necessary that they be zero because $x \otimes x$ cannot be made arbitrary. Therefore, we will proceed on the basis of a sufficiency condition, as proving that the condition is necessary is beyond the scope of this work. Since one solution is for $B_i = 0$, we have

$$\begin{aligned} &2\, Q_{02}\, \pi_s\, (K_2 \otimes 1_u) + 3\, Q_{03}\, \pi_s\, (K_1 \otimes K_1 \otimes 1_u) + 2\, Q_{12}\, [1_x \otimes \pi_s\, (K_1 \otimes 1_u)] + \\ &Q_{21} + V_2\, [(A_{11} \otimes 1_x)S_{21}^1 + (1_x \otimes A_{11})] + 2\, V_2[(A_{02} \otimes 1_x)S_{12}^1 + (1_x \otimes A_{02})] \\ &\quad [1_x \otimes \pi_s\, (K_1 \otimes 1_u)] + V_3\, [(A_{01} \otimes 1_x^{\,2})S_{21}^2 + (1_x \otimes A_{01} \otimes 1_x)S_{21}^1 + \\ &\quad\quad (1_x^{\,2} \otimes A_{01})] = 0 \quad . \end{aligned} \tag{3.35}$$

It is possible to make similar arguments for the remainder of the coefficient terms that multiply $x^m \otimes \Delta u$, $m = 3,4,\ldots,$ if the tensor product is expressed in matrix form as was done for $m=1,2$ here. For higher order cases, it is only possible at the present time to show the sufficiency condition.

Thus far, we have only considered the partial derivative equation of the optimal control $u^*(x,t)$. We also must require that [20]

ORIGINAL PAGE IS
OF POOR QUALITY

$$\frac{d}{dt} V(x,t) + L(x,u^*(x,t),t) = 0 \tag{3.36}$$

along the optimal trajectory. If we collect terms that multiply like tensor powers of x in equation (3.36), we obtain the following expression:

$$\begin{aligned}
&\{\dot{V}_2 + V_2\{(A_{10} \oplus A_{10}) + [(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_1)\} + Q_{02}(K_1 \otimes K_1) + \\
&Q_{11}\,(1_x \otimes K_1) + Q_{20}\}x^2 + \{\dot{V}_3 + V_3\{[(A_{01} \otimes 1_x{}^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + \\
&(1_x{}^2 \otimes A_{01})](1_x{}^2 \otimes K_1) + (A_{10} \oplus A_{10} \oplus A_{10})\} + V_2\{[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})] \\
&(1_x \otimes K_2) + [(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})](1^2_x \otimes K_1) + [(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})] \\
&(1_x \otimes K_1 \otimes K_1) + [(A_{20} \otimes 1_x) + (1_x \otimes A_{20})]\} + Q_{02}[(K_1 \otimes K_2) + (K_2 \otimes K_1)] + \\
&Q_{11}\,(1_x \otimes K_2) + Q_{03}\,(K_1 \otimes K_1 \otimes K_1) + Q_{12}\,(1_x \otimes K_1 \otimes K_1) + Q_{21}\,(1_x{}^2 \otimes K_1) \\
&\qquad + Q_{30}\}\, x^3 + \ldots = 0 \quad .
\end{aligned} \tag{3.37}$$

Figures 3.3 and 3.4 provide a summary of the origins of the terms that comprise equation (3.37). We observe that there are no coefficient terms that multiply the first power of x, hence we begin with the second tensor power. The reader will note that we have collected all of the coeffient terms that multiply like tensor powers of x. In order for the expression (3.37) to be identically zero, it is a sufficient condition that all of the coefficient terms that multiply like tensor powers of x must vanish. The condition has not been shown to be a necessary one because of the fact that the product

$$x \otimes x$$

does not span the space

$$X \otimes X \ .$$

Again, the proof of this being a necessary condition will not be consid-

Original term	Second order terms (those that are multiplied by x^2)	Third order terms (those that are multiplied by x^3)
$Q_{02}\ u^2$	$Q_{02}\ (K_1 \otimes K_1)$	$Q_{02}\ [(K_1 \otimes K_2) + K_2 \otimes K_1)]$
$Q_{20}\ x^2$	Q_{20}	....
$Q_{11}\ x \otimes u$	$Q_{11}\ (1_x \otimes K_1)$	$Q_{11}\ (1_x \otimes K_2)$
$Q_{03}\ u^3$	....	$Q_{03}\ (K_1 \otimes K_1 \otimes K_1)$
$Q_{12}\ x \otimes u^2$	....	$Q_{12}\ (1_x \otimes K_1 \otimes K_1)$
$Q_{21}\ x^2 \otimes u$	....	$Q_{21}\ (1_x \otimes 1_x \otimes K_1)$
$Q_{30}\ x^3$	....	Q_{30}
.	.	.
.	.	.
.	.	.
.	.	.

Figure 3.3

ORIGINAL PAGE IS
OF POOR QUALITY

Original term	Second order terms (those that multiply x^2)	Third order terms (those that multiply x^3)
$\dot{V}_2 x^2$	$\dot{V}_2$	....
$V_2[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})] x \otimes u$	$V_2[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_1)$	$V_2[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})] (1_x \otimes K_2)$
$V_2(A_{10} \oplus A_{10}) x^2$	$V_2 (A_{10} \oplus A_{10})$	....
$V_2[(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})] x^2 \otimes u$	....	$V_2[(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})](1_x^2 \otimes K_1)$
$V_2[(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})] x \otimes u^2$	....	$V_2[(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})](1_x \otimes K_1 \otimes K_1)$
$V_2[(A_{20} \otimes 1_x) + (1_x \otimes A_{20})]$	....	$V_2[(A_{20} \otimes 1_x) + (1_x \otimes A_{20})]$
$\vdots$	$\vdots$	$\vdots$
$\dot{V}_3 x^3$	....	$\dot{V}_3$
$V_3[(A_{01} \otimes 1_x^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1_x^2 \otimes A_{01})]x^2 \otimes u$	....	$V_3[(A_{01} \otimes 1_x^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x) S^1_{21} + (1_x^2 \otimes A_{01})](1_x^2 \otimes K_1)$
$V_3(A_{10} \oplus A_{10} \oplus A_{10}) x^3$	....	$V_3(A_{10} \oplus A_{10} \oplus A_{10})$

Figure 3.4

ORIGINAL PAGE IS
OF POOR QUALITY

ered here due to the scope of the work. This is basically the same question that was found to exist when we collected terms that multiplied $x^m \otimes \Delta u$, $m \geq 2$. At the moment, it is believed that the use of the symmetric tensor algebra will address adequately this issue. Proceeding , then, on the sufficiency criterion, this requirement provides the following conditions:

$$\dot{V}_2 + V_2\{(A_{10} \oplus \text{[illegible]}) + [(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_1)\} + Q_{02}\,(K_1 \otimes K_1) + Q_{11}\,(1_x \otimes K_1) + Q_{20} = 0 \tag{3.38}$$

and

$$\begin{aligned}\dot{V}_3 &+ V_3\{[(A_{01} \otimes 1^2_x)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1^2_x \otimes A_{01})](1^2_x \otimes K_1) + \\ &(A_{10} \oplus A_{10} \oplus A_{10})\} + V_2\,[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_2) + \\ &[(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})](1^2_x \otimes K_1) + [(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})] \\ &(1_x \otimes K_1 \otimes K_1) + [(A_{20} \otimes 1_x) + (1_x \otimes A_{20})]\} + Q_{02}[(K_1 \otimes K_2) + (K_2 \otimes K_1)] + \\ &Q_{11}\,(1_x \otimes K_2) + Q_{03}\,(K_1 \otimes K_1 \otimes K_1) + Q_{12}\,(1_x \otimes K_1 \otimes K_1) + Q_{21}\,(1^2_x \otimes K_1) \\ &+ Q_{30} = 0\,.\end{aligned} \tag{3.39}$$

By considering the previous conditions shown in (3.33) and (3.35) together with the above requirements, it is possible to observe the recursive nature of the problem. The objective is, of course, to solve for the optimal controller gains $K_j(t)$, $j = 1,2,3,\ldots$. We can solve (3.33) for $K_1(t)$ in terms of $V_2(t)$, and substitute this expression into equation (3.38), yielding a differential equation that can be solved for $V_2(t)$ with boundary condition $V_2(t_1) = M_2$, since $V_2(t)$ will be the only unknown. After $V_2(t)$ has been obtained, we can easily obtain the expression for $K_1(t)$. A similar procedure can be employed for the paired terms of $V_3(t)$ and $K_2(t)$, namely, solve equation (3.35) for $K_2(t)$ in terms of $V_3(t)$ and substitute this expression into (3.39), yielding a differential equation that can be solved for $V_3(t)$ with boundary conditions $V_3(t_1) = M_3$. The controller term $K_2(t)$ can then easily be obtained. This procedure can continue until a sufficient number of controller terms are obtained. For the purposes of the example problem that will be considered in the following section, we shall assume that we are only interested in calculating $K_1(t)$ and $K_2(t)$ although the procedure could be extended to yield higher order terms if necessary. The recursive nature of this problem is evident whenever we consider the various term dependencies. It can be concluded from the algorithm that the term $V_{k+1}(t)$ can be determined by knowing only the terms

$$K_1(t),\ K_2(t),\ldots,K_{k-1}(t) \quad \text{and} \quad V_2(t),\ V_3(t),\ldots,V_k(t)$$

and the system description, which means that the controller term $K_k(t)$ can be obtained by knowing only

$$K_1(t),\ K_2(t),\ldots,K_{k-1}(t) \quad \text{and} \quad V_2(t),\ V_3(t),\ldots,V_{k+1}(t)\ .$$

This result was presented in [1]. In the next section of this chapter we show that the LQ problem which has been extensively studied in optimal regulation theory is equivalent to the first set of solutions obtained with the methods presented here, that is, the set of equations that provide $V_2(t)$ and $K_1(t)$.

3.3 The LQ Problem

In this section, we wish to show that the LQ problem is obtained from our results as a special case, namely the case that results from the truncation of the system description to linear terms and the performance index to quadratic terms. This result will verify the equation that

ORIGINAL PAGE IS
OF POOR QUALITY

was derived in the previous section, which is the equation that expresses the relationship between $K_1(t)$ and $V_2(t)$ (3.33).

The normal formulation of the LQ problem has a linear system description

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

and a quadratic $L(x,u,t)$ in the performance index

$$J = \frac{1}{2} M(x(t_1)) + \frac{1}{2} \int_{t_0}^{t_1} L(x,u,t)dt ,$$

where

$$L(x,u,t) = x^T Qx + u^T Ru + x^T Cu, \; M(x_f) = x^T(t_1) Mx(t_1) .$$

The reader will note the inclusion of the cross term $x^T Cu$. Although not usually considered in the classical literature, our methods will treat this term as being quadratic as well. It can be shown by using the classical approach to optimization problems that the unique optimal control $u(t)$ can be expressed as

$$u(t) = K_1(t)\; x(t) ,$$

where

$$K_1(t) = -\, R^{-1} \,(B^T\, V(t) + \frac{1}{2}\, C^T) . \tag{3.40}$$

The matrix $V(t)$ is known to satisfy the differential equation

$$\begin{aligned} &\dot{V} + V(A - \frac{1}{2}\, B\, R^{-1}\, C^T) + (A^T - \frac{1}{2}\, C\, R^{-1}\, B^T)\, V - \\ &V\, B\, R^{-1}\, B^T\, V + Q - \frac{1}{4}\, C\, R^{-1}\, C^T = 0, \; V(t_1) = M . \end{aligned} \tag{3.41}$$

If we were to assume that $C = 0$, which is typical for most practical problems, we obtain the usual matrix form of the Riccati equation, which is

$$\dot{V} + VA + A^T V - VBR^{-1}B^T V + Q = 0 . \tag{3.42}$$

ORIGINAL PAGE IS
OF POOR QUALITY

Our objective in this section will be to demonstrate that the equations produced by the expressions previously derived yield the identical equations produced by the LQ problem as considered here, namely equations (3.33) and (3.38). We begin with our expression relating K_1 and V_2, which is

$$V_2[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})] + Q_{11} + 2\, Q_{02}\, \pi_s\, (K_1 \otimes 1_u) = 0\ . \qquad (3.43)$$

We shall assume that, for dim U = dim X = 2,

$$A_{01} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

and

$$K_1 = \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}\ .$$

If

$$R = \begin{bmatrix} r_1 & r_2 \\ r_2 & r_3 \end{bmatrix}$$

this implies that

$$Q_{02} = [r_1\ r_2\ r_2\ r_3]\ .$$

Likewise, if

$$C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

this implies that

$$Q_{11} = [c_{11}\ c_{12}\ c_{21}\ c_{22}]\ .$$

Also, if

$$V = \begin{bmatrix} V_{21} & V_{22} \\ V_{22} & V_{24} \end{bmatrix}$$

ORIGINAL PAGE IS
OF POOR QUALITY

this implies that

$$V_2 = [V_{21}\ V_{22}\ V_{22}\ V_{24}] \ .$$

It is easily verified that the expression (3.43) produces the following set of equations, assuming that the V matrix is symmetric, which follows from the quadratic nature of the solution $x^T Vx$. We have

$$2r_1\ E_{11} + 2r_2\ E_{21} + 2\ b_{11}\ V_{21} + 2\ b_{21}\ V_{22} + c_1 = 0 \ ,$$
$$2r_1\ E_{12} + 2r_2\ E_{22} + 2\ b_{11}\ V_{22} + 2\ b_{21}\ V_{24} + c_3 = 0 \ ,$$
$$2r_2\ E_{11} + 2r_3\ E_{21} + 2\ b_{12}\ V_{21} + 2\ b_{22}\ V_{22} + c_2 = 0 \ ,$$
$$2r_2\ E_{12} + 2r_3\ E_{22} + 2\ b_{12}\ V_{22} + 2\ b_{22}\ V_{24} + c_4 = 0 \ .$$

If the equations produced by the classical solution (3.40) are compared with these, one would find that the equations are identical.

The other part of the LQ problem involves the verification of the Riccati equation (3.41) beginning with equation (3.38), which is

$$\dot{V}_2 + V_2\{(A_{10} \oplus A_{10}) + [(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_1)\} +$$
$$Q_{02}\ (K_1 \otimes K_1) + Q_{11}\ (1_x \otimes K_1) + Q_{20} = 0 \ ,$$

where

$$A_{10} \triangleq A$$

and

$$A_{01} \triangleq B \ .$$

As was previously discussed, the relationship between the Q_{02}, Q_{11}, and Q_{20} vectors and the classical weighting matrices R, C, and Q involves the assumption of an ordering convention on the various basis elements. Since we are assuming that the basis elements are to be ordered lexicographically, the Q_{ij} vectors can easily be obtained from these classical weighting matrices.

It can be observed that the expressions (3.41) and (3.38) are very similar. The proof that identical equations are produced is not difficult but is very tedious when done with the generalized matrices. Therefore, for the time being, we will assume that these two expressions provide identical equations. When the example problem is considered in the following section, we will verify that the derived expression (3.38) produces results identical to the classical Riccati equation.

3.4 Discussion

In this section, we have derived the necessary equations in order to calculate the first and second order controller gains and have presented the method by which controller gains of any desired order could be calculated. Because of the various term dependencies, we showed the recursive nature of the solution, namely the alternating solution of the terms in the expansion for the optimal value function and the terms in the controller gain expansion. The LQ problem was partially demonstrated to be a special case of the derived algorithm, namely the solution for $V_2(t)$ and $K_1(t)$ for generalized matrices from the dimension 2 case. At this point, an example is needed to solidify the concepts presented so far and demonstrate the method of solution of the recursive expressions.

IV. APPLICATION TO SPECIFIC EXAMPLE

In this section, we present one of the major contributions of this work - application of the concepts presented thus far to an example problem with a complete set of calculations. The particular example that will be considered was analyzed extensively in [9] with particular emphasis on system modeling and model following with a variety of excitation functions. Because the models generated were proven to closely approximate the true solution, these models will be well-suited for our purposes. We will present the ideas and techniques for constructing a nonlinear control for this particular example based on the expressions derived in the previous sections. Basically, we will calculate both a first order and a second order control and present an analysis of the resulting equations for each case.

4.1 Problem Requirements and Formulation

Before we begin the process of specifying an example and choosing appropriate weighting matrices that appear in the cost functional, it will be advisable to consider the requirements that we wish to meet in formulating a meaningful example problem. We will list some of the criteria which we have considered in this choice.

(1) The eigenvalues of A_{10} should be in the left half plane (that is, they should have real parts less than zero); the idea here is that gas turbine engine models are typically stable.

(2) The choice $(x(t),u(t)) \equiv 0$ is a solution to the differential equation.

(3) The Q_{20} and M_2 terms are not zero simultaneously.

ORIGINAL PAGE IS
OF POOR QUALITY

(4) the composite matrix

$$\left[\begin{array}{c|c} Q & \frac{1}{2}C \\ \hline \frac{1}{2}C^T & R \end{array}\right]$$

is positive definite.

In order to clarify the controllability issues, we must first assume that the time interval is some finite time t_1. The optimal feedback system will turn out to be time varying as long as the control interval is finite. This will turn out to be the case even when the system and cost functionals are time-invariant, which shall be assumed for the example system. The engineering construction of these time-varying functions can easily be done on the digital computer using standard integration techniques. It can be shown [1,22] that if we let $t_1 \to \infty$, then we obtain a time-invariant controller for a time-invariant system. Basically, controllability is required here in order to ensure that the cost is finite. Using a finite interval, there are several examples noted in [22] where an optimal control is obtained for an uncontrollable system. The requirement that the Q_{20} and M_2 terms are not zero simultaneously excludes the trivial case that would produce the optimal control $u(t) \equiv 0$, although these terms could be zero individually.

Now that the example requirements have been formulated, we are ready to choose an example system and demonstrate how the gain matrices may be calculated for the first and second order cases. The particular example that we will choose was effectively studied and analyzed in [9] with particular emphasis on nonhomogeneous model following. We consider the system

$$f_1\ (x,u) = \dot{x}_1 = u_2 \cosh\ (x_1 x_2) - e^{2u_1} \sinh\ (2x_1) - 3 \sinh\ (x_2)\ ,$$
$$f_2\ (x,u) = \dot{x}_2 = e^{u_1 u_2} \sinh\ (x_1) - e^{u_1} u_1 \cosh\ (x_1{}^2) + \sinh\ (x_2)\ ,$$

and a performance index

$$J = \frac{1}{2} M\ (x(t_1)) + \frac{1}{2} \int_{t_0}^{t_1} L(x,u,t)\ dt$$

where we choose the following values for the weighting terms Q_{ij} and M_j:

$$Q_{20} = [2\ 0\ 0\ 2]\ ,$$
$$Q_{02} = [5\ 0\ 0\ 5]\ ,$$
$$Q_{11} = [6\ 0\ 0\ 6]\ ,$$
$$M_2 = [1\ 0\ 0\ 2]\ ,$$
$$M_3 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]\ .$$

We note that the Q_{ij} terms were chosen such that the requirement (4) is satisfied, which is necessary for a meaningful example problem. Also, we are assuming that the penalty term

$$M(x(t_1)) = M(x_f)$$

is required to be a convex nonnegative function which means that $M(x_f)$ is specified in terms of even powers of x_f [1].

Since the A_{pq} operators are not directly available, we must use the concepts presented in Section 3.1 where these operators were observed to be related to various partial derivative matrices, namely

$$A_{10} = \left.\begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} \\[2ex] \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} \end{bmatrix}\right|_{\substack{x = (0,0) \\ u = (0,0)}}$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$= \begin{bmatrix} -2 & -3 \\ 1 & 1 \end{bmatrix}$$

and

$$A_{01} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} \end{bmatrix}\Bigg|_{\substack{x = (0,0) \\ u = (0,0)}}$$

$$= \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} .$$

We can easily observe that the requirements (1) and (2) have been satisfied; the origin in $X \times U$ defines a solution, since

$$f_i\ (0,0) = 0 ,$$

for $i = 1,2$, and A_{10} has eigenvalues with negative real parts, which assures local stability when u is zero. The remainder of the system matrices can be similarly calculated as was done for A_{10} and A_{01}, as follows:

$$A_{11} = \begin{bmatrix} \frac{\partial f_1^{\,2}}{\partial x_1 \partial u_1} & \frac{\partial f_1^{\,2}}{\partial x_1 \partial u_2} & \frac{\partial f_1^{\,2}}{\partial x_2 \partial u_1} & \frac{\partial f_1^{\,2}}{\partial x_2 \partial u_2} \\ \frac{\partial f_2^{\,2}}{\partial x_1 \partial u_1} & \frac{\partial f_2^{\,2}}{\partial x_1 \partial u_2} & \frac{\partial f_2^{\,2}}{\partial x_2 \partial u_1} & \frac{\partial f_2^{\,2}}{\partial x_2 \partial u_2} \end{bmatrix}\Bigg|_{\substack{x = (0,0) \\ u = (0,0)}}$$

$$= \begin{bmatrix} -4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$A_{20} = \frac{1}{2}\left[\begin{matrix} \frac{\partial^2 f_1}{\partial x_1^2} & \frac{\partial^2 f_1}{\partial x_1 \partial x_2} & \frac{\partial^2 f_1}{\partial x_2 \partial x_1} & \frac{\partial^2 f_1}{\partial x_2^2} \\ \frac{\partial^2 f_2}{\partial x_1^2} & \frac{\partial^2 f_2}{\partial x_1 \partial x_2} & \frac{\partial^2 f_2}{\partial x_2 \partial x_1} & \frac{\partial^2 f_2}{\partial x_2^2} \end{matrix}\right]\Bigg|_{\substack{x = (0,0) \\ u = (0,0)}}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and

$$A_{02} = \frac{1}{2}\left[\begin{matrix} \frac{\partial^2 f_1}{\partial u_1^2} & \frac{\partial^2 f_1}{\partial u_1 \partial u_2} & \frac{\partial^2 f_1}{\partial u_2 \partial u_1} & \frac{\partial^2 f_1}{\partial u_2^2} \\ \frac{\partial^2 f_2}{\partial u_1^2} & \frac{\partial^2 f_2}{\partial u_1 \partial u_2} & \frac{\partial^2 f_2}{\partial u_2 \partial u_1} & \frac{\partial^2 f_2}{\partial u_2^2} \end{matrix}\right]\Bigg|_{\substack{x = (0,0) \\ u = (0,0)}}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}.$$

These methods provide only the nominal or analytical values of the A_{pq} system matrices and may be unsatisfactory due to the truncation of some of the higher order terms. A more accurate method employs the least-squares minimization technique using the Singular Value Decomposition, which is extensively discussed in [9]. Basically, the results obtained for the system matrices via least squares minimization provided much more accurate responses than those obtained with the standard linear approximation and provided a very close match to the "true" solution. In order to test and verify this model, various input signals were employed with different frequencies and amplitudes. In all of these cases, it was found that the model performed remarkably well, and should be well-

ORIGINAL PAGE IS
OF POOR QUALITY

suited for the generation of the various controller gains which we will accomplish in the next section. For the particular example that we have chosen, it is discussed in [9] that a degree-2 model (that is, a model that is generated by the methods of identification and contains only the system matrices A_{10}, A_{01}, A_{20}, A_{11}, and A_{02}) is adequate to effectively describe the system as noted above. The identification and verification of a degree-3 model involved more effort and programming time and was found to only slightly outperform the degree-2 models, and then only for a fraction of the test points. Using the least squares identification methods, the following matrices were obtained as the degree-2 model for the example system:

$$A_{10} = \begin{bmatrix} -2.001 & -3.009 \\ 1.006 & 1.011 \end{bmatrix},$$

$$A_{01} = \begin{bmatrix} 0.002 & 0.997 \\ -1.000 & 0.000 \end{bmatrix},$$

$$A_{11} = \begin{bmatrix} -4.150 & -0.074 & -0.048 & -0.176 \\ -0.007 & 0.083 & 0.008 & 0.102 \end{bmatrix},$$

$$A_{20} = \begin{bmatrix} 0.239 & -0.0725 & -0.0725 & -0.720 \\ -0.323 & -0.064 & -0.064 & 0.359 \end{bmatrix},$$

$$A_{02} = \begin{bmatrix} -0.105 & 0.0135 & 0.0135 & 0.012 \\ -0.982 & 0.0075 & 0.0075 & -0.013 \end{bmatrix}.$$

We observe that the A_{10} and A_{01} operators closely approximate the nominal values obtained by partial differentiation. By using these

ORIGINAL PAGE IS
OF POOR QUALITY

values, we can begin to calculate the required terms in the expansion for the nonlinear controller.

4.2 Calculation of Controller Terms

In this section, we show how to calculate the terms of the nonlinear controller expansion. For the sake of simplicity, we will only demonstrate the methods to obtain the gain matrices $K_1(t)$ and $K_2(t)$, along with the optimal value expansion terms $V_2(t)$ and $V_3(t)$. As previously mentioned, equations for the calculation of higher order terms could be developed using the same methods that provided the equations for $K_1(t)$ and $K_2(t)$ if these terms were desired. The first step in the procedure is the solution of equation (3.33) for K_1 in terms of V_2. We have

$$2\, Q_{02}\, \pi_s\, (K_1 \otimes 1_u) + Q_{11} + V_2\, [(A_{01} \otimes 1_x)S_{11}^1 + (1_x \otimes A_{01})] = 0 \ .$$

If we make the necessary substitutions, that is, assume that

$$Q_{02} = [5\ 0\ 0\ 5] \ ,$$

$$\pi_s = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ,$$

$$K_1 = \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix} ,$$

$$Q_{11} = [6\ 0\ 0\ 6] \ ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$S_{11}^{1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and use the previously noted values for the A_{01} system matrix, we find that we can express the components of K_1 by the following expression:

$$\begin{bmatrix} E_{11} \\ E_{12} \\ E_{21} \\ E_{22} \end{bmatrix} = \begin{bmatrix} -.0004 & .2 & 0 \\ -.0004 & 0 & .2 \\ -.1994 & 0 & 0 \\ 0 & -.1994 & 0 \end{bmatrix} \begin{bmatrix} V_{21} \\ V_{22} \\ V_{24} \end{bmatrix} + \begin{bmatrix} -.6 \\ 0 \\ 0 \\ -.6 \end{bmatrix} . \qquad (4.1)$$

In obtaining this equation, we have assumed that $V_{22} = V_{23}$, which corresponds to assuming that the V matrix from classical optimal control theory is symmetric. A quick comparison to the equation (3.40) reveals that our results are indeed correct.

The next step in the algorithm is the substitution of equation (4.1) into the equation that was obtained by collecting terms that multiply x^2 when the HJB equation is evaluated along the optimal trajectory. This result is equation (3.38), which is

$$\dot{V}_2 + V_2 \{(A_{10} \oplus A_{10}) + [(A_{01} \otimes 1_x)S_{11}^{1} + (1_x \otimes A_{01})] (1_x \otimes K_1)\} +$$
$$Q_{02} (K_1 \otimes K_1) + Q_{11} (1_x \otimes K_1) + Q_{20} = 0 .$$

If we make this substitution for K_1 and use our assumed values for the system and performance index matrices, we can obtain the following set of coupled differential equations which can be solved for V_2 using numerical methods.

ORIGINAL PAGE IS
OF POOR QUALITY

$$\dot{V}_{21} - 4.044\ V_{21} + 3.212\ V_{22} + .1988\ V_{21}^2 - .0008\ V_{21}\ V_{22} + .2\ V_{22}^2 + .2 = 0\ ,$$

$$\dot{V}_{22} - 3.6072\ V_{21} - 1.011\ V_{22} + 1.606\ V_{24} + .1988\ V_{21}\ V_{22} - .0004\ V_{22}^2 - .0004\ V_{21}\ V_{24} + .2\ V_{21}\ V_{24} = 0\ ,$$

$$\dot{V}_{24} - 7.2144\ V_{22} + 2.022\ V_{24} + .1988\ V_{22}^2 - .0008\ V_{22}\ V_{24} + .2\ V_{24}^2 + .2 = 0\ . \qquad (4.2)$$

As a check on the validity of these equations, we will compare the equations listed in (4.2) to those obtained by the Riccati equation, which was described in (3.41) as

$$\dot{V} + V\ (A - \frac{1}{2}\ B\ R^{-1}\ C^T) + (A - \frac{1}{2}\ B\ R^{-1}\ C^T)^T\ V - V\ B\ R^{-1}\ B^T\ V + Q - \frac{1}{4}\ C\ R^{-1}\ C^T = 0\ .$$

After substituting the appropriate values for the known quantities, we obtain the following set of differential equations for $V(t)$.

$$\dot{V}_{21} - 4.044\ V_{21} + 3.212\ V_{22} + .1988\ V_{21}^2 - .0008\ V_{21}\ V_{22} + .2\ V_{22}^2 + .2 = 0\ ,$$

$$\dot{V}_{22} - 3.6072\ V_{21} - 1.011\ V_{22} + 1.606\ V_{24} + .1988\ V_{21}\ V_{22} - .0004\ V_{22}^2 - .0004\ V_{21}\ V_{24} + .2\ V_{21}\ V_{24} = 0\ , \qquad (4.3)$$

$$\dot{V}_{24} - 7.2144\ V_{22} + 2.022\ V_{24} + .1988\ V_{22}^2 - .0008\ V_{22}\ V_{24} + .2\ V_{24}^2 + .2 = 0\ .$$

These differential equations are to be solved with the boundary conditions

$$V_{21}\ (t_1) = 1\ ,$$

$$V_{22}\ (t_1) = 0\ ,$$

and

$$V_{24}\ (t_1) = 2\ .$$

Since the differential equations presented in (4.2) agree with the classical Riccati solutions in (4.3), we may conclude that our results are indeed correct.

C-2

The solution of the first order control problem involves two basic steps. In the first step, the set of equations (4.3) are solved for $V_2(t)$ and subsequently $K_1(t)$ by using numerical integration from the final value of $V_2(t)$. The second part of the procedure involves calculations for the control matrix $K_1(t)$, which then follows from the known matrix values of $V_2(t)$, on the interval of solution. For this first step of the calculations, two approaches may be taken, either a solution in terms of tensor quantities, or a solution taking advantage of the well known procedures for Riccati equations. Since we have previously shown that our first order results are identical to those produced by the classical Riccati equation, it is used in the first order analysis program, which is called FIRORDA [23,24]. This program is listed in Appendix G of this report. Numerical values for K_1 and V_2 appear also in this appendix, denoted by KA and VA.

In order to test briefly the behavior of the software for the first order feedback, the values of the weighting matrices that comprise the performance index were changed to five items their assumed values, and these results compared to those produced with the example values assumed for these matrices. As the values for these matrices were increased, we found that the solution $V_2(t)$ of the Riccati equation became more of a time varying gain, with greater initial values than those provided by the example problem case. This comparison can be made by referring to Figures 4.1 and 4.2. In obtaining these arrays, we have somewhat arbitrarily assumed an interval of $t = 0$ to $t = 5$ seconds, with an integration stepsize of 0.1 seconds. These results generally agree with the results presented in [22]. Note in these two figures that V_2 is denoted by VA.

ORIGINAL PAGE IS
OF POOR QUALITY

VA (A 51 BY 5 ARRAY)				
0	.18131	.16595	.16595	.52464
.1	.18193	.166	.166	.52432
.2	.1828	.16639	.16639	.5242
.3	.18389	.1672	.1672	.5245
.4	.18513	.16847	.16847	.52545
.5	.18643	.17017	.17017	.52727
.6	.18765	.17219	.17219	.53005
.7	.18867	.17436	.17436	.53381
.8	.18935	.17643	.17643	.53839
.9	.18962	.17812	.17812	.54349
1	.18947	.17915	.17915	.54867
1.1	.18896	.17932	.17932	.55339
1.2	.18829	.17854	.17854	.55709
1.3	.18773	.17687	.17687	.5593
1.4	.18767	.17462	.17462	.55976
1.5	.1865	.17226	.17226	.55851
1.6	.1906	.17049	.17049	.556
1.7	.19428	.17009	.17009	.5531
1.8	.19965	.17185	.17185	.55109
1.9	.2066	.17642	.17642	.55154
2	.21477	.18417	.18417	.5561
2.1	.22351	.19504	.19504	.56627
2.2	.23198	.20845	.20845	.58307
2.3	.23921	.22328	.22328	.60681
2.4	.24428	.23792	.23792	.63679
2.5	.2465	.25043	.25043	.67123
2.6	.24556	.2588	.2588	.70726
2.7	.2418	.26129	.26129	.7412
2.8	.23626	.25688	.25688	.76[illegible]
2.9	.2[illegible]75	.24561	.24561	.78666
3	.22781	.22898	.22898	.79167
3.1	.23038	.21013	.21013	.78328
3.2	.24145	.19367	.19367	.76363
3.3	.26352	.18535	.18535	.73812
3.4	.29802	.19126	.19126	.71541
3.5	.34482	.21687	.21687	.70679
3.6	.40192	.26597	.26597	.72496
3.7	.46546	.33968	.33968	.78236
3.8	.52991	.4358	.4358	.88946
3.9	.5886	.54837	.54837	1.0529
4	.63446	.66767	.66767	1.2739
4.1	.66087	.78051	.78051	1.5467
4.2	.66284	.87085	.87085	1.8573
4.3	.63842	.92107	.92107	2.1824
4.4	.59058	.91414	.91414	2.4887
4.5	.52945	.83734	.83734	2.7343
4.6	.47448	.68783	.68783	2.873
4.7	.45545	.47976	.47976	2.8652
4.8	.5106	.25071	.25071	2.6929
4.9	.68065	.063532	.063532	2.3791
5	1	0	0	2

Figure 4.1 $V_2(t)$ for Example Performance Index

ORIGINAL PAGE IS
OF POOR QUALITY

VA (A 51 BY 5 ARRAY)

0	.89611	.82271	.822[illegible]	2.5[illegible]
.1	.89677	.82246	.82246	2.59[illegible]
.2	.89776	.82275	.82275	2.5917
.3	.89905	.8237	.8237	2.5911
.4	.90049	.82533	.82533	2.5915
.5	.90[illegible]	.82759	.82759	2.5931
.6	.[illegible]0563	.83025	.83025	2.5961
.7	.90378	.83298	.83298	2.6005
.8	.90379	.83536	.83536	2.606
.9	.90299	.83692	.83692	2.6[illegible]7
1	.90136	.8372	.8372	2.6174
1.1	.89906	.83587	.83587	2.62[illegible]
1.2	.89641	.83283	.8328[illegible]	2.6236
1.3	.89387	.82825	.82825	2.6227
1.4	.89205	.82265	.82265	2.6164
1.5	.89159	.81687	.81687	2.6108
1.6	.89306	.81204	.81204	2.[illegible]08
1.7	.89686	.80944	.80944	2.5[illegible]99
1.8	.90306	.81027	.81027	2.5802
1.9	[illegible]1141	.81549	.81549	2.5741
2	.[illegible]111	.82551	.82551	2.57[illegible]
2.1	[illegible]	.84	.84	2.[illegible]
2.2	.[illegible]759	.85777	.85777	[illegible]
2.3	.[illegible]525	.87673	.87673	2.[illegible]
2.4	[illegible]1646	.89405	.89405	2.[illegible]7
2.5	[illegible]4214	.90647	.90647	2.7047
2.6	[illegible]	.91076	.91076	2.[illegible]
2.7	[illegible]	.90435	.90435	2.[illegible]7
2.8	.[illegible]7855	.88597	.88597	2.[illegible]
2.9	.[illegible]7786	.85625	.85625	2.7921
3	.[illegible]6485	.81812	.81812	2.7672
3.1	.[illegible]3629	.77695	.77695	2.7189
3.2	.[illegible]6423	.7402	.7402	2.6[illegible]16
3.3	.[illegible]3584	.71661	.71661	2.5[illegible]
3.4	.92437	.71495	.71495	2.50[illegible]3
3.5	.[illegible]7645	.74243	.74243	2.4491
3.6	1.0436	.80301	.80301	2.4357
3.7	1.1125	.89588	.89588	2.4772
3.8	1.1752	1.0144	1.0144	2.5846
3.9	1.2207	1.146	1.146	2.7603
4	1.[illegible]38	1.2723	1.2723	2.9961
4.1	1.2185	1.3713	1.3713	3.2[illegible]13
4.2	[illegible]	1.4198	1.4198	3.5535
4.3	1.0585	1.3973	1.3973	3.7993
4.4	.9368	1.2905	1.2905	3.9611
4.5	.79412	1.0986	1.0986	3.993
4.6	.57588	.83691	.83691	3.8614
4.7	.[illegible]2665	.53967	.53967	3.5557
4.8	[illegible]771	.2585	.2585	3.098
4.9	[illegible]1151	.05662	.05662	2.5479
5	1	0	0	2

Figure 4.2 $V_2(t)$ for Scaled Performance Index

ORIGINAL PAGE IS
OF POOR QUALITY

A similar procedure is used for the second order analysis of the example problem as was used for the first order analysis. Namely, since the values for $K_1(t)$ and $V_2(t)$ are known from the first order analysis problem, equation (3.34) can be solved for $K_2(t)$ in terms of $V_3(t)$. This equation is

$$\begin{aligned}
&2\, Q_{02}\, \pi_s\, (K_2 \otimes 1_u) + V_2\, [(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})] + \\
&2\, V_2\, [(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})][1_x \otimes \pi_s\, (K_1 \otimes 1_u)] + \\
&V_3\, [(A_{01} \otimes 1^2_x)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1^2_x \otimes A_{01})] = 0\ . \qquad (4.4)
\end{aligned}$$

We note that we have assumed $Q_{03} = Q_{12} = Q_{21} = 0$ for the purpose of the example problem, that is, higher order than quadratic terms in the expansion for $L(x,u,t)$ are zero. Since equation (4.4) cannot be solved directly for $K_2(t)$ in terms of $V_3(t)$, we must multiply the first term in the expansion.

It should be pointed out that the manner in which K_2 is intertwined into (4.4) is one of the most interesting features of the derivation in this report, which avoids as much as feasible the use of symmetric algebra and dual spaces. In future work, we hope to return to this point.

Another interesting feature of (4.4) can be seen in more than one place, but perhaps especially in the coefficient of V_3 on the third line. Notice that this coefficient consists of three terms, each in essence a rearrangement of A_{01} and two copies of 1_x. Symmetric algebra would simplify such sums.

Returning to the first term in (4.4), we have

ORIGINAL PAGE IS
OF POOR QUALITY

$$2\,Q_{02}\,\pi_s\,(K_2 \otimes 1_u) =$$

$$2\,[r_1\ r_2\ r_2\ r_3]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} F_{11} & 0 & F_{12} & 0 & F_{13} & 0 & F_{14} & 0 \\ 0 & F_{11} & 0 & F_{12} & 0 & F_{13} & 0 & F_{14} \\ F_{21} & 0 & F_{22} & 0 & F_{23} & 0 & F_{24} & 0 \\ 0 & F_{21} & 0 & F_{22} & 0 & F_{23} & 0 & F_{24} \end{bmatrix}$$

for

$$K_2 = \begin{bmatrix} F_{11} & F_{12} & F_{13} & F_{14} \\ F_{21} & F_{22} & F_{23} & F_{24} \end{bmatrix}.$$

Multiplying these expressions, we have

$$2\,Q_{02}\,\pi_s\,(K_2 \otimes 1_u) = [2r_1F_{11} + 2r_2F_{21},\ 2r_2F_{11} + 2r_3F_{21},\ 2r_1F_{12} + 2r_2F_{22},$$
$$2r_2F_{12} + 2r_3F_{22},\ 2r_1F_{13} + 2r_2F_{23},\ 2r_2F_{13} + 2r_3F_{23},\ 2r_1F_{14} + 2r_2F_{24},$$
$$2r_2F_{14} + 2r_3F_{24}]$$

$$= [F_{11}\ F_{12}\ F_{13}\ F_{14}\ F_{21}\ F_{22}\ F_{23}\ F_{24}]\begin{bmatrix} 2r_1 & 2r_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2r_1 & 2r_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2r_1 & 2r_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2r_1 & 2r_2 \\ 2r_2 & 2r_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2r_2 & 2r_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2r_2 & 2r_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2r_2 & 2r_3 \end{bmatrix}$$

$$= \hat{K}_2\,G\ ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

where $\hat{K}_2$ denotes the values of K_2 rearranged into an 8-vector and G is the 8x8 matrix above. Using this notation, we can solve for $\hat{K}_2$ in terms of V_3 in equation (4.3) as

$$\hat{K}_2 = - \{V_2[(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})] + 2\, V_2[(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})][1_x \otimes \pi_s(K_1 \otimes 1_u)] + V_3\, [(A_{01} \otimes 1_x{}^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1_x{}^2 \otimes A_{01})]\}\, G^{-1} \tag{4.5}$$

The second order solution can be obtained by substituting from equation (4.5) into the equation

$$\dot{V}_3 + V_3\, \{[(A_{01} \otimes 1_x{}^2)S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1_x{}^2 \otimes A_{01})](1_x{}^2 \otimes K_1) + (A_{10} \oplus A_{10} \oplus A_{10})\} + V_2\, \{[(A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01})](1_x \otimes K_2) + [(A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11})](1_x{}^2 \otimes K_1) + [(A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02})](1_x \otimes K_1 \otimes K_1) + [(A_{20} \otimes 1_x) + (1_x \otimes A_{20}]\} + Q_{02}\, [(K_1 \otimes K_2) + (K_2 \otimes K_1)] + Q_{11}\, (1_x \otimes K_2) = 0\ ,$$

which is equation (3.39) with the third and higher order terms in the expansion for $L(x,u,t)$ equated to zero. The method of solution of the second order control problem is slightly more complicated than the first order solution, but employs the same basic steps.

4.3 Discussion

In this section, we have examined in terms of an explicit example the computational requirements associated with the computation of higher order nonlinear feedback control terms based upon the use of tensor descriptions. Preliminary experience with such computations has been encouraging. Appendix G contains a listing of SECORDA, the program for K_2 and V_3, whose values appear there also, as K2A and V3A.

A major purpose of this example study has been to assess the merits of using the symmetric tensor algebra and the concepts of dual spaces in the calculation of nonlinear feedback tensor gains. To this end, we have employed such ideas as infrequently as possible. The results have been revealing. For the most part, one can bypass the ideas in question and substitute matrix algebra notions such as transpose. Interestingly enough, however, there are perhaps two or three instances, one in (4.4), where one is strongly inclined to consider the insertion of more technicality.

With this observation in mind, we are making a study of the computational requirements involved in [1], where symmetry and duals are both used, in order to determine tradeoffs with the current study.

V. CONCLUSIONS

In the three sections preceding, we have carefully examined the computational implications of [1], for the case of an example system which had received a rather thorough study in a prior grant year. Here we have to point out that [1] makes use of technical tensor machinery involving mixed tensors on spaces and their duals, the theory of contractions, and the formal use of symmetric tensor algebra. The formal purpose of this study has been to assess the computational implications of such nonlinear feedback control theory, and in particular to examine the possibility of suppressing the explicit use of dual spaces and contractions. One benefit, for example, of such a suppression would be an avoidance of the distinction between tensors and vector-valued tensors. It has been numerically clear for a long time, as explained in earlier reports, that the implicit use of symmetry was going to be a certainty. A question, however, was whether the explicit use of symmetry in the algebraic derivations would really be necessary.

Having been as careful as we could to carry out the derivation of this report while using such concepts as little as possible, we have made some important conclusions. First, we had to dip into formal symmetric notation for a brief segment of our derivation, in connection with certain differentiations. Whether this was an absolute necessity or not tends to be outweighed by the fact that a great deal of effort would be required in order to circumvent it. We conclude that the formal use of symmetry should be pursued. While this does result in another level of equivalence relations, the terminology and symbolism can be induced without a new order of magnitude of difficulty. It tends to be more a question of replacing one tensor language with another tensor language. Second, we have encountered steps in the equations where duality may be a great help in unraveling a complicated twist of symbols. At this point, we assess this ability as important, so that we are initiating a careful look at the way in which these computations would change if we made use of contractions. Should this work out with the expected benefits, it may serve as a very motivating engineering example of the practical importance of duality. While the role of duality in general optimization theory has been known for many years, there are many common instances in which it can be finessed by standard matrix terminology. Such an example could be, therefore, very compelling.

In addition to the work presented in the body of this report, the appendices contain a number of items which we also believe to be milestones. Appendices B and E deal with ongoing progress in the computer software required for tensor modeling and simulation. This developing package is the outgrowth of years of work and for the first time makes it possible to do examples on a reasonable time scale. In particular, plans are underway to use this new capability to enhance the efforts on tensor model identification and order reduction. We expect to make a more complete report on this package within the next six months. Appendix C introduces a new mathematical viewpoint on the nonlinear feedback control synthesis problem. Basically, the idea is to redefine the vector space structures for the inputs and outputs of nonlinear systems. For important cases, this can be done in such a way that the nonlinear system becomes a linear system with respect to the new spaces. We have this idea under study, and have already shown that it greatly expands the possibilities for defining comparison sensitivity functions on nonlinear cases. We envision the use of such sensitivity functions to aid in characterizing the quality of order reductions and model identifications. In Appendix D we continue our investigation into the role of tensors in controller scheduling. We expect that the procedures of scheduling may contain one of the keys in resolving the tradeoff questions involving nonlinear model dynamical quality versus useful region. Finally, in Appendix F, we present a recent compilation of references on design by total synthesis, a sequence of studies which trace some of their roots back to the first years of this grant.

VI. REFERENCES

1. M.R. Buric, "Optimal Regulation of Nonlinear Systems by Methods of Tensor Algebra", Ph.D. Dissertation, Department of Electrical Engineering, University of Minnesota, Minneapolis, Minnesota, March, 1978.

2. E.G. Al'brekht, "On the Optimal Stabilization of Nonlinear Systems", Journal of Applied Mathematics and Mechanics, 25, 1962.

3. D.L. Lukes, "Optimal Regulation of Nonlinear Dynamic Systems", SIAM Journal of Control, Volume 7, No. 1, 1969.

4. W.L. Garrard, N.H. McClamroch, and L.G. Clark, "An Approach to Sub-Optimal Feedback Control of Nonlinear Systems", International Journal of Control, 5, 1967.

5. W.L. Garrard, "Additional Results on Sub-Optimal Feedback Control of Nonlinear Systems", International Journal of Control, 10, 1969.

6. W.L. Garrard and J.M. Jordan, "Design of Nonlinear Automatic Flight Control Systems", Automatica, Volume 13, 1977.

7. E.B. Lee and L. Markus, Foundations of Optimal Control Theory, New York: Wiley and Sons, 1967.

8. V. Seshadri, "Multivariable Feedback Loop Closures: An Approach by Exterior Algebra", Ph.D. Dissertation, Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, January, 1980.

9. S. Yurkovich, "Application of Tensor Ideas to Nonlinear Modeling and Control", M.S.E.E. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, January, 1981.

10. M.K. Sain, "The Growing Algebraic Presence in Systems Engineering: An Introduction", Proceedings of the IEEE, Volume 64, No. 1, pp. 96-111, January, 1976.

11. M. Marcus, Finite Dimensional Multilinear Algebra, parts I and II, New York: Dekker, 1975.

12. H. Wielandt, Finite Permutation Groups, New York: Academic Press, 1964.

13. R. Bellman, Introduction to Matrix Analysis, New York: McGraw-Hill, 1960.

14. G. Rogers, Matrix Derivatives, New York: Dekker, 1980.

15. M.K. Sain, Introduction to Algebraic System Theory, New York: Academic Press, 1981.

16. W.H. Greub, Linear and Multilinear Algebra, parts I and II, New York: Springer, 1967.

17. A. Sagle and R. Walde, Introduction to Lie Groups and Lie Algebras, New York: Academic Press, 1973.

18. J. Dieudonne, Foundations of Modern Analysis, New York: Academic Press, 1960.

19. S. Yurkovich, Research Status Notes, January 25, 1982 (to be published in a Ph.D. dissertation at a later date at the University of Notre Dame).

20. J.C. Hsu and A.U. Meyer, Modern Control Principles and Applications, New York: McGraw-Hill, 1968.

21. John W. Brewer, "Kronecker Products and Matrix Calculus in System Theory", IEEE Transactions on Circuits and Systems, Volume CAS-25, Number 9, September, 1978.

22. M. Athans and P.L. Falb, Optimal Control, New York: McGraw-Hill, 1966.

23. S. Cohen and S.C. Pieper, The Speakeasy III Reference Manual, Chicago: Speakeasy Computing Corporation, 1979.

24. W. Jennings, First Course in Numerical Methods, New York: The Macmillan Company, 1964.

25. W.L. Garrard and L.G. Clark, "On the Synthesis of Suboptimal, Inertia-Wheel Altitude Control Systems", Automatica, 5, 1969.

26. C.K. Ling, "Quasi-Optimum Design of an Aircraft Landing Control System", J. Aircraft, Jan.-Feb., 1970.

27. R.C. Durbeck, "An Approximation Technique for Suboptimal Control", IEEE Transactions on Automatic Control, AC-10, 1965.

28. B. Carnahan, et al, Applied Numerical Methods, New York: Wiley, 1979.

APPENDIX A

GRANT BIBLIOGRAPHY, INCEPTION TO PRESENT

1. R.J. Leake, J.L. Melsa and M.K. Sain, "Preliminary Proposal on Modern Approaches to Jet Engine Control", November 1, 1974.

2. R.J. Leake, M.K. Sain and J.L. Melsa, "Proposal to NASA for Support of a Work Entitled 'Alternatives for Jet Engine Control'", November 27, 1974.

3. R.J. Leake and M.K. Sain, "Semi-Annual Status Report on NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1975-August 31, 1975.

4. R.J. Leake, M.K. Sain and J.L. Melsa, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 1, 1975.

5. J.C. Shearer, "An IBM 370/158 Installation and User's Guide for the DYNGEN Jet Engine Simulator", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, November 1975.

6. T.C. Brennan, "Simplified Simulation Models for Control Studies of Turbojet Engines", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, January 1976; also published as Technical Report 757, November 1975.

7. R.J. Leake, M.K. Sain and J.L. Melsa, "Final Technical Report on NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1975 - February 29, 1976.

8. R.R. Gejji, "Polynomial Techniques Applied to Multivariable Control", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1976; also published as Technical Report EE 761, March 1976.

9. V. Seshadri, "Compensation of Multivariable Control Systems", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana; May 1976; also published as Technical Report EE 763, May 1976.

10. A.J. Maloney, III, "Graphics Analysis of Dominance in Jet Engine Control Models", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, August 1976; also published as Technical Report EE 765, June 24, 1976.

11. M.K. Sain, R.J. Leake, R. Basso, R. Gejji, A. Maloney and V. Seshadri, "Alternative Methods for the Design of Jet Engine Control Systems", Proceedings Fifteenth Joint Automatic Control Conference, pp. 133-142, July 1976.

12. J.C. Shearer, W.E. Longenbaker, Jr. and R.J. Leake, "An IBM 370/158 Installation and User's Guide to the DYNGEN Jet Engine Simulator", Technical Report EE 766, July 1976.

13. R.R. Gejji and M.K. Sain, "A Jet Engine Control Problem for Evaluating Minimal Design Software", Proceedings Midwest Symposium on Circuits and Systems, pp. 238-243, August 1976.

14. V. Seshadri and M.K. Sain, "Interaction Studies on a Jet Engine Model by Characteristic Methodologies", Proceedings Midwest Symposium on Circuits and Systems, pp. 232-237, August 1976.

15. R. Basso and R.J. Leake, "Computational Alternatives to Obtain Time-Optimal Jet Engine Control", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, August 1976; also published as Technical Report EE 767, September 1976.

16. R.J. Leake and M.K. Sain, "Semi-Annual Status Report on NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1976-August 31, 1976.

17. R. Basso and R.J. Leake, "Computational Methods to Obtain Time-Optimal Jet Engine Control", Proceedings Fourteenth Allerton Conference on Circuit and System Theory, pp. 652-661, September 1976.

18. M.K. Sain and V. Seshadri, "A Result on Pole Assignment by Exterior Algebra", Proceedings Fourteenth Allerton Conference on Circuit and System Theory, pp. 399-407, September 1976.

19. R.R. Gejji, "A Computer Program to Find the Kernel of a Polynomial Operator", Proceedings Fourteenth Allerton Conference on Circuit and System Theory, pp. 1091-1100, September 1976.

20. R.J. Leake and M.K. Sain, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 25, 1976.

21. R.M. Schafer, W.E. Longenbaker and M.K. Sain, "System Dominance: A Preliminary Report on an Alternate Approach", Technical Report, University of Notre Dame, November 25, 1976.

22. R.J. Leake and M.K. Sain, "Final Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1976-February 28, 1977.

23. R.M. Schafer, "A Graphical Approach to System Dominance", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1977; also published as Technical Report EE 772, April 1, 1977.

24. R.J. Leake, J.G. Allen and R.S. Schlunt, "Optimal Regulators and Follow-Up Systems Determined from Input-Output Signal Monitoring", Proceedings IEEE International Symposium on Circuits and Systems, April 1977.

25. W.E. Longenbaker and R.J. Leake, "Hierarchy of Simulation Models for a Turbofan Gas Engine", Proceedings Eighth Annual Pittsburgh Conference on Modeling and Simulation, April 1977.

26. P.W. Hoppner, "The Direct Approach to Compensation of Multivariable Jet Engine Models", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1977; also published as Technical Report EE 774, May 1977.

27. R.R. Gejji and R.J. Leake, "Time-Optimal Control of a Single Spool Turbojet Engine Using a Linear Affine Model", Technical Report EE 7711, University of Notre Dame, June 1977.

28. R.M. Schafer, R.R. Gejji, P.W. Hoppner, W.E. Longenbaker and M.K. Sain, "Frequency Domain Compensation of a DYNGEN Turbofan Engine Model", Proceedings Sixteenth Joint Automatic Control Conference, pp 1013-1018, June 1977.

29. R.R. Gejji and M.K. Sain, "Application of Polynomial Techniques to Multivariable Control of Jet Engines", Proceedings Fourth IFAC Symposium on Multivariable Technological Systems, pp. 421-429, July 1977.

30. R.R. Gejji, "Use of DYGABCD Program at Off-Design Points", Technical Report EE 7703, University of Notre Dame, July 1977.

31. E.A. Sheridan and R.J. Leake, "Non-Interactive State Request Jet Engine Control with Non-Singular B Matrix", Proceedings Twentieth Midwest Symposium on Circuits and Systems, pp. 539-543, August 1977.

32. R. Gejji, R.M. Schafer, M.K. Sain and P. Hoppner, "A Comparison of Frequency Domain Techniques for Jet Engine Control System Design", Proceedings Twentieth Midwest Symposium on Circuits and Systems, pp. 680-685, August 1977.

33. R.J. Leake and M.K. Sain, "Semi-Annual Status Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1977 -August 31, 1977.

34. W.E. Longenbaker, "Time Optimal Control of a Two-Spool Turbofan Jet Engine", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, January 1978; also published as Technical Report 7714, September 1977.

35. R.J. Leake and J.G. Comiskey, "A Direct Method for Obtaining Nonlinear Analytical Methods of a Jet Engine", Proceedings International Forum on Alternatives for Linear Multivariable Control, National Electronics Conference, Chicago, pp. 203-212, October 1977.

36. M.K. Sain and R.J. Leake, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 25, 1977.

37. J.A. Ortega and R.J. Leake, "Discrete Maximum Principle with State Constrained Control", SIAM Journal on Control and Optimization, Vol. 15, No. 6, pp. 109-115, November 1977.

38. Michael K. Sain and V. Seshadri, "Pole Assignment and a Theorem from Exterior Algebra", Proceedings IEEE Conference on Decision and Control, pp. 291-295, December 1977.

ORIGINAL PAGE IS
OF POOR QUALITY

39. R. Michael Schafer and Michael K. Sain, "Some Features of CARDIAD Plots for System Dominance", Proceedings IEEE Conference on Decision and Control, pp. 801-806, December 1977.

40. M.K. Sain, "The Theme Problem", in Alternatives for Linear Multivariable Control, M.K. Sain, J.L. Peczkowski and J.L. Melsa, Editors. Chicago: National Engineering Consortium, 1978, pp. 20-30.

41. R.M. Schafer and M.K. Sain, "Input Compensation for Dominance of Turbofan Models", in Alternatives for Linear Multivariable Control, M.K. Sain, J.L. Peczkowski, and J.L. Melsa, Editors. Chicago: National Engineering Consortium, 1978, pp. 156-169.

42. J.L. Peczkowski and M.K. Sain, "Linear Multivariable Synthesis with Transfer Functions", in Alternatives for Linear Multivariable Control, M.K. Sain, J.L. Peczkowski, and J.L. Melsa, Editors. Chicago: National Engineering Consortium, 1978, pp. 71-87.

43. R.J. Leake and M.K. Sain, "Final Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1977 -February 28, 1978.

44. M.K. Sain and R.J. Leake, "Semi-Annual Status Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1978 -August 31, 1978.

45. R.J. Leake, J.L. Peczkowski and M.K. Sain, "Step Trackable Linear Multivariable Plants", Technical Report EE 789, September 1978.

46. R.M. Schafer and M.K. Sain, "CARDIAD Design: Progress in the Four Input/Output Case", Proceedings Sixteenth Allerton Conference on Communication, Control, and Computing, p. 567, October 1978.

47. M.K. Sain, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", November 1, 1978.

48. M.K. Sain, "On Exosubsets and Internal Models", Proceedings IEEE Conference on Decision and Control, pp. 1069-1073, January 1979.

49. M.K. Sain, "Annual Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1978 -February 28, 1979.

50. V. Seshadri and M.K. Sain, "An Application of Exterior Algebra to Multivariable Feedback Loops", Proceedings Conference on Information Sciences and Systems, Johns Hopkins University, pp. 337-342, 1979.

51. M.K. Sain and R.M. Schafer, "Alternatives for Jet Engine Control", Propulsion Controls Symposium, NASA Lewis Research Center, Preprints, p. III-15, May 1979.

52. J.G. Comiskey, "Time Optimal Control of a Jet Engine Using a Quasi-Hermite Interpolation Model", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1979; also published as Technical Report EE 791, May 1979.

53. R.M. Schafer and M.K. Sain, "Frequency Dependent Precompensation for Dominance in a Four Input/Output Theme Problem Model", Proceedings Eighteenth Joint Automatic Control Conference, pp. 348-353, June 1979.

54. M.K. Sain, "Semi-Annual Status Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1979-August 31, 1979.

55. V. Seshadri and M.K. Sain, "Loop Closures and the Induced Exterior Map", Proceedings Seventeenth Allerton Conference on Communications, Control, and Computing, pp. 753-761, October 1979.

56. R. Michael Schafer and Michael K. Sain, "CARDIAD Approach to System Dominance with Application to Turbofan Engine Models", Proceedings Thirteenth Asilomar Conference on Circuits, Systems, and Computers, pp. 78-82, November 1979.

57. R.J. Leake, J.L. Peczkowski and M.K. Sain, "Step Trackable Linear Multivariable Plants", International Journal of Control, Vol. 30, pp. 1013-1022, December 1979.

58. R.R. Gejji, "On the Total Synthesis Problem of Linear Multivariable Control", Ph.D. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1980; also published as Technical Report EE 801, January 1980.

59. V. Seshadri, "Multivariable Feedback Loop Closures: An Approach by Exterior Algebra", Ph.D. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, January 1980; also published as Technical Report EE 800, January 1980.

60. M.K. Sain, "Semi-Annual Status Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", September 1, 1979-February 29, 1980.

61. R.M. Schafer, "On the Design of Dynamical Compensation for Diagonal Dominance", Ph.D. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, May 1980.

62. M.K. Sain, A. Ma and D. Perkins, "Sensitivity Issues in Decoupled Control Systems Design", Proceedings Twelfth Southeastern Symposium on System Theory, pp. 25-29, May 1980.

63. S. Yurkovich and M. Sain, "Generating Nonlinear Models from Digital Simulations: A Tensor Approach", Proceedings Eleventh Pittsburgh Conference on Modeling and Simulation, pp. 797-801, May 2, 1980.

64. M.K. Sain, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", June 19, 1980.

65. R.R. Gejji, "Reliable Floating Point Computation of Minimal Bases", Proceedings Nineteenth Joint Automatic Control Conference, Paper WA-8B, August 1980.

66. M.K. Sain, "Nineteen Month Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", March 1, 1979 -September 30, 1980.

67. S. Yurkovich and M. Sain, "A Tensor Approach to Modeling of Nonhomogeneous Nonlinear Systems", Proceedings Eighteenth Allerton Conference on Communications, Control, and Computing, pp. 604-613, October 1980.

68. M.K. Sain, "Quotient Signal Flowgraphs: New Insights", Proceedings IEEE International Conference on Circuits and Computers, p. 417, October 1980.

69. Stephen Yurkovich, "Application of Tensor Ideas to Nonlinear Modeling and Control", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, January 1981.

70. M.K. Sain, "Semi-Annual Status Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 1, 1980 -March 31, 1981.

71. V. Seshadri, "Exterior Algebra and Simultaneous Pole-Zero Placement", Proceedings Conference on Information Sciences and Systems, Johns Hopkins University, 1981.

72. J.L. Peczkowski and M.K. Sain, "Scheduled Nonlinear Control Design for a Turbojet Engine", Proceedings IEEE International Symposium on Circuits and Systems, pp. 248-251, April 1981.

73. M.K. Sain, "Status of Computer-Aided Control System Feedback Design: Algebraic Methods", GE-RPI-NSF Workshop on Control Design, May 20, 1981.

74. Stephen Yurkovich, Thomas A. Klingler and Michael K. Sain, "Tensor Ideas for Nonlinear Modeling of a Turbofan Jet Engine: Preliminary Studies", Proceedings Twelfth Annual Pittsburgh Conference on Modeling and Simulation, pp. 1423-1427, May 1981.

75. M.K. Sain, "Proposal for Continuation of NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", June 11, 1981.

76. M.K. Sain and J.L. Peczkowski, "An Approach to Robust Nonlinear Control Design", Proceedings Joint Automatic Control Conference, FA-3D, June 1981.

77. M.K. Sain, "Twelve Month Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 1, 1980 -September 30, 1981.

78. M.K. Sain, "Semi-Annual Status Report on NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 1, 1981-March 31, 1982.

79. T.A. Klingler, S. Yurkovich and M.K. Sain, "An Application of Tensor Ideas to Nonlinear Modeling of a Turbofan Jet Engine", Proceedings Thirteenth Pittsburgh Conference on Modeling and Simulation, April 1982.

80. J.P. Hill, "Solution of Nonlinear Optimal Control Problems Using the Algebraic Tensor: An Example", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, June 1982.

81. M.K. Sain and R.M. Schafer, "A Computer-Assisted Approach to Total Feedback Synthesis", Proceedings American Control Conference, pp. 195-196, June 1982.

82. M.K. Sain and J.L. Peczkowski, "Nonlinear Multivariable Design by Total Synthesis", Proceedings American Control Conference, pp. 252-260, June 1982.

83. M. Sain and S. Yurkovich, "Controller Scheduling: A Possible Algebraic Viewpoint", Proceedings American Control Conference, pp. 261-269, June 1982.

84. T.A. Klingler, "Nonlinear Modeling of a Turbofan Jet Engine: A Tensor Method Application", M.S. Thesis, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, August 1982.

85. T.A. Klingler, S. Yurkovich and M.K. Sain, "A Computer-Aided Design Package for Nonlinear Model Applications", Preprints Second IFAC Symposium on Computer Aided Design of Multivariable Technological Systems, pp 345-353, September 1982.

86. M.K. Sain, "Twelve Month Technical Report, NASA Grant NSG-3048, 'Alternatives for Jet Engine Control'", October 1, 1981 -September 30, 1982.

APPENDIX B

"An Application of Tensor Ideas to Nonlinear Modeling of a Turbofan Jet Engine"

T.A. Klingler
S. Yurkovich
M.K. Sain

Proceedings Thirteenth Pittsburgh Conference on Modeling and Simulation

April 1982

ORIGINAL PAGE IS
OF POOR QUALITY

AN APPLICATION OF TENSOR IDEAS TO
NONLINEAR MODELING OF A TURBOFAN JET ENGINE*

Thomas A. Klingler, Stephen Yurkovich, and Michael K. Sain
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556

ABSTRACT

The design of nonlinear control systems for gas turbine engines frequently involves a combination of feedforward scheduling and local, dynamic feedback regulation on the desired final responses. Scheduling the feedback dynamics, or adding dynamical tuning to the feedforward schedules, creates a class of nonlinear dynamical controllers which is often classical in nature, as for example the first few terms in a series expansion. Tensor algebra provides a universal setting within which to parameterize such representations. Moreover, if such models are available for the engine itself, then there exist feedback control theories based upon them. In this paper, a model of tensor type is computed and tested locally on a digital simulation of the QCSE gas turbine engine.

INTRODUCTION

The use of local, linear dynamical models in control of gas turbine engines has received a great deal of attention in the last ten years. While the lion's share of control action for such engines tends to be the result of feedforward schedules, the local feedback applied to reach desired response points along these schedules is of great importance. In particular, careful choice of the local controller dynamics can achieve quick, smooth settling, without undesirable overshoots in crucial variables, as for example temperatures in the vicinity of turbines.

Such local dynamics are frequently scheduled also, as a function of a smoothly changing physical variable, such as a speed. When this is accomplished, the local control dynamics become nonlinear in nature; and key examples can be viewed in terms of vector fields created by polynomic functions of state and control, or, more generally, in terms of power series. Tensor algebra provides a universal parameterization within which to represent such schemes. Moreover, there exist feedback theories designed to accommodate plant models based upon such representations.

Accordingly, there is interest in application studies of tensor models. In this paper, we provide one such study, on a QCSE engine simulator.

For background, we consider briefly some tensor ideas and issues associated with nonlinear modeling. A short description of the QCSE engine itself is given, and then the application is discussed in detail.

TENSOR IDEAS

We begin our discussion with a brief description of the tools to be employed in the nonlinear model formulation. Let V and W be real vector spaces and let $(\otimes^r V, \otimes^r)$ be a tensor product for r copies of V, where each integer r is two or greater. For convenience we define $\otimes^1 V = V$ and $\otimes^0 = R$. Then by the unique factorization property of the tensor product [1], for every r-linear mapping

$$\psi : V^r \to W \tag{1}$$

there exists a unique linear mapping

*This work was supported by the National Aeronautics and Space Administration under Grant NSG 3048.

ORIGINAL PAGE IS
OF POOR QUALITY

$$\lambda : \otimes^r V \to W \tag{2}$$

such that $\psi = \lambda \circ \otimes^r$ for the r-linear mapping $\otimes^r : V^r \to \otimes^r V$. If $L(V^r;W)$ denotes the real vector space of r-linear mappings from V^r to W, and $L(\otimes^r V,W)$ denotes the real vector space of linear mappings from $\otimes^r V$ to W, the implication is that

$$L(\otimes^r V,W) \to L(V^r;W) \tag{3}$$

is a vector space isomorphism.

These notions may be tied to the discussion of abstract derivatives and the calculus on normed vector spaces. As an introduction, equip V and W with norms and let Z be open in V. Suppose that the mapping $f : Z \to W$ is differentiable at a point p in Z, in the usual sense (see, for example, [2,3]). We denote the derivative of $f : Z \to W$ at p by

$$(Df)(p) : V \to W, \tag{4}$$

and note that

$$Df : Z \to L(V,W); \tag{5}$$

that is, the derivative mapping (4) is a linear mapping, an element of $L(V,W)$. The notion extends for higher derivatives, defined in a recursive fashion as

$$(D^r f)(p) = (D(D^{r-1} f))(p) \tag{6}$$

provided the $(r-1)$st derivative is differentiable, since

$$\begin{aligned} D^2 f(p) &\in L(V,L(V,W)), \\ D^3 f(p) &\in L(V,L(V,L(V,W))), \end{aligned} \tag{7}$$

and so on. It can be shown that there exist isomorphisms

$$\begin{aligned} L(V^2;W) &\to L(V,L(V,W)), \\ L(V^3;W) &\to L(V,L(V,L(V,W))), \\ &\vdots \qquad\qquad \vdots \end{aligned} \tag{8}$$

so that $D^r f(p)$ can be regarded as an r-linear mapping $V^r \to W$, up to isomorphism. We suppress this isomorphism and think of $D^r f(p)$ as just such a mapping.

It is now straightforward to establish a connection with the tensor ideas expressed above. The r-linear mapping (1), for our purposes given by $D^r f(p) : V^r \to W$, can be composed from a linear mapping $\otimes^r V \to W$ and the universal r-linear tensor product mapping $\otimes^r : V^r \to \otimes^r V$. This connection, facilitated by the isomorphisms (3) and (8), is explored in the section following for the case of dynamical system representation.

MODEL STRUCTURE

Suppose that the dynamical system which we wish to model is described by the nonlinear ordinary differential equation

$$\dot{x} = f(x,u) \tag{9}$$

for $f : X \times U \to X$, where X and U are normed real vector spaces of states and controls, respectively. Using the notation of the preceding section, let $(\bar{x},\bar{u})$ be a fixed point in Z open in $X \times U$, and suppose that $f : X \times U \to X$ is of sufficient smoothness on Z. Then, formally,

$$f(\bar{x} + x,\ \bar{u} + u) = \sum_{k=0}^{\infty} \frac{1}{k!} (D^k f)(\bar{x},\bar{u})(x,u)^{(k)}, \tag{10}$$

where $(x,u)^{(k)} = ((x,u),(x,u),\ldots,(x,u))$ k times. We note that the series in (10) could be represented by a finite number of terms together with a remainder term in a standard application of Taylor's formula. Indeed, for practical applications, such as the present paper, a truncation approximation of (10) is considered. Unfortunately, limitations of space forbid discussions concerning such issues as existence of solutions to (9) or questions related to the convergence of (10).

ORIGINAL PAGE IS
OF POOR QUALITY

We now make use of the fact that $(D^k f)(\bar{x},\bar{u})$ in (10) is a k-linear mapping, which suggests a means of applying tensor product ideas. Let $(\otimes^k(X \times U),\otimes^k)$ be a tensor product for k copies of $X \times U$. Then we may make the unique factorization

$$D^k f(\bar{x},\bar{u}) = L_k(\bar{x},\bar{u}) \cdot \otimes^k, \tag{11}$$

where $L_k(\bar{x},\bar{u}) : \otimes^k(X \times U) \to X$ is a linear mapping. Now let the notation $(x,u)^k$ denote the k-fold tensor product of (x,u) with itself. Then upon substitution of (11) into (10) we have

$$f(\bar{x}+x,\bar{u}+u) = \sum_{k=0}^{\infty} \frac{1}{k!} L_k(\bar{x},\bar{u})(x,u)^k. \tag{12}$$

It is shown in [4] that the individual terms of (12) may be rewritten as, for example,

$$\frac{1}{2!} L_2(\bar{x},\bar{u})(x,u)^2 = L_{20}(\bar{x},\bar{u})x\otimes x + L_{11}(\bar{x},\bar{u})x\otimes u + L_{02}(\bar{x},\bar{u})u\otimes u. \tag{13}$$

In this way the formal expansion (10) becomes

$$f(\bar{x}+x,\bar{u}+u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} L_{ij}(\bar{x},\bar{u})x^i \otimes u^j, \tag{14}$$

which forms the structure for the nonlinear model.

As alluded to earlier, in practice the series (14) may be truncated in an approximation of (9). The task in the model building scheme, then, is to identify the parameters contained in matrix representations of the $L_{ij}(\bar{x},\bar{u})$, once ordered bases for the spaces in question are chosen. For more discussion of the details involved in such an exercise, the reader may wish to consult [5,6].

QCSE ENGINE

The intent of this section is to supply a brief introduction to NASA's QCSEE ("Quixie")---Quiet, Clean, Shorthaul Experimental Engine---prior to discussing an application of the modeling methodology described above. The QCSE engine is an advanced turbofan designed specifically for powered-lift, short-haul aircraft, and combines several innovative concepts to achieve optimal efficiency with quiet, clean operation [7,8]. The eight physical quantities chosen as state variables for the system include two fan speeds, four pressures, and two temperatures. A digital controller is incorporated into the overall design [9], and the control inputs are the main burner fuel flow, the fan pitch angle, and the fan nozzle area.

For the modeling exercises of this study, a detailed digital simulation developed for the QCSE engine [10] is employed. The primary input variable to be manipulated in the digital program is the percentage power demand, PWRX, for testing performance over the entire envelope of operation. Values of individual internal variables are extracted and inserted at various locations within the program.

APPLICATION

Attention in the following discussion will center around the formulation of a reduced order four-state, three-control analytical model. The engine states chosen are the combustor discharge pressure (P4GS), the core nozzle total pressure (P8GS), and the rotor dynamics in the form of fan speed (NL), and compressor speed (NH). All three engine control inputs are employed, namely, the main burner fuel flow (WFM), the exhaust nozzle area (A18) and the fan pitch angle (BETAF).

Appropriate engine operation for the model identification involves opening the loop by deactivating the controller and independently inserting the individual control inputs. In this strategy, nonlinearities of the plant exist--which might otherwise be less noticeable had the controller been present in the loop. A further explanation of this strategy as well as an alternate one are presented in [11]. An important point to note is that, in the open loop situation, the choice of input control signals is critical. This is due to the fact that the engine itself has certain physical limits, which in turn have been incorporated into the simulator. In reality, exceeding these limits could cause severe damage to the engine, an example of which is turbine melt down.

To aid in the selection of input signals, a family of parametric plots have been constructed using QCSEE steady state data from idle (62.5% PWRX) to maximum power (100% PWRX). Figure 1 contains an example of one such steady state plot. From these plots a set of acceptable input signals can be selected. Acceptable state perturbations can be selected in a similar

ORIGINAL PAGE IS
OF POOR QUALITY

fashion. Another important feature of these steady state plots is that they suggest regions of nonlinearity. From Figure 1 it appears that in the locality of 92% power demand the engine is nonlinear due to the abrupt changes in exhaust nozzle area and fan pitch angle. With this in mind, we shall establish 92% as the operating point of the present study. Model formulations at other operating points are currently under investigation.

The following is an overview of the identification procedure. The QCSEE simulator is run, closed loop, with a 92% power demand for ten seconds to settle all transients. This produces the equilibrium value $(\bar{x},\bar{u})$, where $\bar{x}$ is a four-tuple and $\bar{u}$ is a three-tuple. The initial conditions thus generated form the point of expansion for the series truncation approximation in the model formulation. Within the digital simulation program the controller is disconnected by setting the control derivatives to zero. From the steady state plots a point $(\hat{x},\hat{u})$ is chosen on the engine operating line at 93% power demand. The state variables are perturbed x from their equilibrium values where

$$x = \hat{x} - \bar{x}.$$

Furthermore, the control variables are manipulated so that a cosinusoidal input of amplitude α is inserted, where α is a three-tuple given by the expression

$$\alpha = \hat{u} - \bar{u}.$$

The observed states and inputs are sampled over a six second interval; 100 samples are evenly spaced at .06 seconds, and the difference between these values and the corresponding equilibrium values, together with the ordered monomials from the tensor product terms (see [5]) comprise one of two blocks of data necessary for the identification. The second bloc of data consists of the state derivative values which are extracted directly from the simulator at the given sample rate. Through use of these data blocks, the parameters contained in matrix representations of the $L_{ij}(\bar{x},\bar{u})$ can be identified via a least squares minimization technique.

Using the above procedure, two models have been identified: a second-degree nonlinear model, and a first-degree linear model. The linear model has been identified for use in comparison studies. The second-degree approximation keeps second degree tensor products which are associated with quadratic terms. Accordingly, such a nonlinear model is expected to outperform the linear model in a region about the point of expansion.

A simple error comparison criterion is used in testing the performance of the nonlinear model versus that of the linear model. Let ε_i^N denote the absolute maximum error in the nonlinear model solution, as compared to the true simulation solution, over the time range of simulation for the ith state variable. Similarly, we define ε_i^L for the linear model error. Then ε_i is the comparison $\varepsilon_i^N - \varepsilon_i^L$. Thus, if ε_i is negative, the nonlinear model has exhibited a smaller maximum absolute error in the ith state, and in that sense has outperformed the linear model. Table 1 contains a list of the state variables, their corresponding QCSEE variable name, their unit of measure, as well as their corresponding state notation x_i. Samples of the error comparison for various initial conditions, input amplitudes and frequencies are presented in Table 2. All input frequencies are in Hertz.

The error criterion in Table 2 clearly indicates that the nonlinear model outperforms the linear model in a region about the equilibrium point; however, there exists a better method for revealing model performance, namely, trajectory comparison. Consequently, a representative number of comparative solution plots have been included in Figures 2-10. Figure 2 offers a simulation of pressure P8GS for a step response, whereas Figures 3-4 illustrate P8GS and NL respectively for the frequency set ϕ = (.25, 0., .5). A simulation of P4GS is shown in Figure 5 for an excursion away from the typical engine line of operation, and likewise Figure 6 depicts NH. Figures 7-8 illustrate the speeds NL and NH for a 1% amplitude control signal, and finally P4GS and NL are seen in Figures 9-10 with initial conditions of approximately 1%.

COMPUTING ENVIRONMENT

The software package, developed using the extensive capabilities of the IBM and DEC Command Procedure Languages and the strengths of FORTRAN and SPEAKEZY, is divided into two segments and tailored to utilize effectively existent computer hardware. The interactive nonlinear model generation segment is implemented on a Time Sharing Option (TSO) of the IBM 370-168 computer system, where the memory dependent, and highly computational routines of the package can benefit from use of the virtual memory and floating point hardware.

Once a nonlinear model is identified on the IBM 370-168, it is transferred to a DEC PDP 11/60, where, in an interative environment, it can be analyzed and compared to a linear model and the true solution. This is accomplished through use of the nonlinear simulation segment

ORIGINAL PAGE IS
OF POOR QUALITY

of the package. In this manner, the routines can use both the graphics capabilities of a Tektronix 4025 video terminal, and a Versatec electrostatic printer/plotter for the display of data and comparative trajectories.

CONCLUSIONS

This paper has presented an application illustration of tensor modeling to a digital simulation of the QCSE engine. For plant modeling prior to feedback control, or for representing scheduled controllers over an operating line, the tensor algebra offers a universal parameterization which is helpful in conceptualization and identification. The case studied in this paper offers support to these conclusions. Further work is in progress.

ACKNOWLEDGEMENT

The authors are pleased to thank Mr. Daniel Bugajski for his assistance in certain of the computer studies associated with these results. Dan is a senior in the Department of Electrical Engineering.

REFERENCES

1. W.H. Greub, Multilinear Algebra. New York: Springer-Verlag, 1967.
2. J. Dieudonne, Foundations of Modern Analysis. New York: Academic Press, 1960.
3. S. Lang, Analysis. Vols. I and II, Reading: Addison-Wesley, 1969.
4. M. Sain and S. Yurkovich, "Controller Scheduling: A Possible Algebraic Viewpoint", Proc. 1982 American Control Conf., June 14-16, 1982, to appear.
5. S. Yurkovich and M.K. Sain, "A Tensor Approach to Modeling of Nonhomogeneous Nonlinear Systems", Proc. Eighteenth Allerton Conf. on Communication, Control, and Computing, October 1980, pp. 604-613.
6. S. Yurkovich, "Application of Tensor Ideas to Nonlinear Modeling and Control", M.S. Thesis, Electrical Engineering Dept., Univ. of Notre Dame, Notre Dame, Indiana, Jan. 1981.
7. C.E. Wise, "Turbofan of the Future", Machine Design, Aug. 22, 1974, pp. 20-25.
8. "Quiet Clean Short-haul Experimental Engine (QCSEE) Under-the-Wing (UTW) Simulation Report", NASA CR-134914, July 1977.
9. "Quiet Clean Short-haul Experimental Engine (QCSEE) Under-the-Wing (UTW) Engine Digital Control System Design Report", NASA CR-134920, January 1978.
10. J.R. Mihaloew, "A Nonlinear Propulsion System Simulation Technique for Piloted Simulators", Proc. Twelfth Pittsburgh Conf. on Modeling and Simulation, April 1981, pp. 1407-1417.
11. S. Yurkovich, T.A. Klingler, and M.K. Sain, "Tensor Ideas for Nonlinear Modeling of a Turbofan Jet Engine: Preliminary Studies", Proc. Twelfth Pittsburgh Conf. on Modeling and Simulation, April 1981, pp. 1423-1427.

TABLE 1 Variable Ledger for Figures 2-10

o : True Engine Response	Δ : Linear Model Response	* : Nonlinear Model Response	
x_1: P4GS (psi) x_2: P8GS (psi)	x_3: NL (rpm) x_4: NH (rpm)	u_1: WFM (lb_m/hr) u_2: A18 (in^2)	u_3: BETAF (degrees)

TABLE 2 Comparison Studies

Initial State Conditions x_1	x_2	x_3	x_4	Input Amplitudes α_1	α_2	α_3	Input Freq. ϕ_1	ϕ_2	ϕ_3	Error ε_1	ε_2	ε_3	ε
0.000	0.000	0.000	0.000	18.92	0.000	-0.111	0.0	0.0	0.0	-1.040	-0.018	-10.40	-14.00
0.000	0.000	0.000	0.000	18.92	0.000	-0.111	0.3	0.0	0.5	-0.668	-0.012	- 7.04	- 8.49
0.000	0.000	0.000	0.000	74.29	-21.20	-0.239	1.9	0.9	1.2	-0.259	-0.004	- 4.62	- 8.11
0.010	0.001	0.010	0.100	17.00	-2.000	-0.050	0.0	0.0	0.0	-0.947	-0.017	- 7.48	-11.50
0.010	0.001	0.010	0.100	18.92	0.000	-0.111	0.0	0.0	0.0	-1.070	-0.019	-10.60	-14.20
0.010	0.001	0.010	0.100	19.00	-2.000	-0.159	1.0	0.8	0.5	-0.198	-0.003	- 2.49	- 3.16
0.010	0.001	0.010	0.500	18.92	0.000	-0.111	0.3	0.0	0.5	-0.642	-0.012	- 6.88	- 8.22
0.010	0.001	-0.010	0.500	40.00	-5.000	-0.150	1.5	1.0	1.3	-0.354	-0.006	- 1.96	- 6.69
0.010	-0.001	0.010	-0.750	-18.80	0.000	0.115	2.0	0.0	1.5	-0.307	-0.005	- 1.71	- 4.89
-0.010	-0.001	-0.010	-0.750	-37.46	0.000	0.228	1.9	0.0	1.3	-0.202	-0.004	- 1.59	- 3.42
-0.010	0.001	-0.010	0.750	74.29	-21.20	-0.239	1.9	1.1	1.3	-0.293	-0.005	- 5.89	- 6.55
0.075	0.001	20.00	10.00	-18.80	0.000	0.000	1.9	0.0	0.0	-0.343	-0.007	- 3.36	- 3.43
0.110	0.001	-20.00	12.00	55.78	-14.03	-0.198	1.8	1.1	1.4	-0.447	-0.008	- 4.98	- 5.51
0.120	0.002	50.00	10.00	37.35	-6.830	-0.139	2.1	1.0	1.3	-0.435	-0.008	- 5.06	- 4.91
0.100	0.002	75.00	5.00	54.00	-12.00	-0.200	1.5	1.0	1.3	-0.401	-0.007	- 9.80	- 8.80

ORIGINAL PAGE IS
OF POOR QUALITY

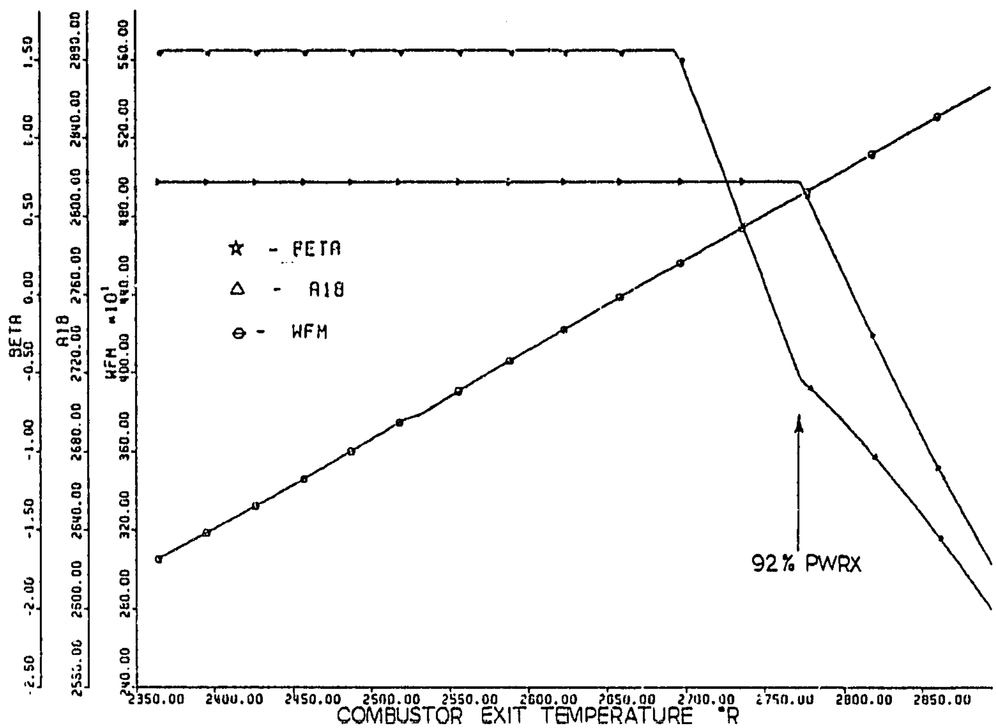

Figure 1 Example Steady State Response: Parameterization with PWRX

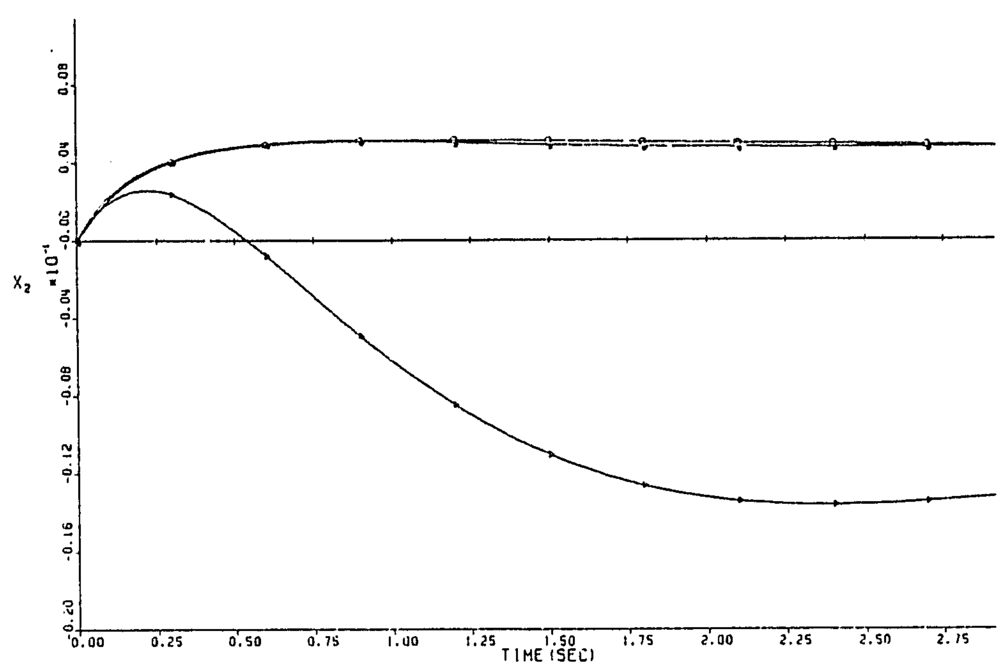

Figure 2 Table 2, Line 1

ORIGINAL PAGE IS OF POOR QUALITY

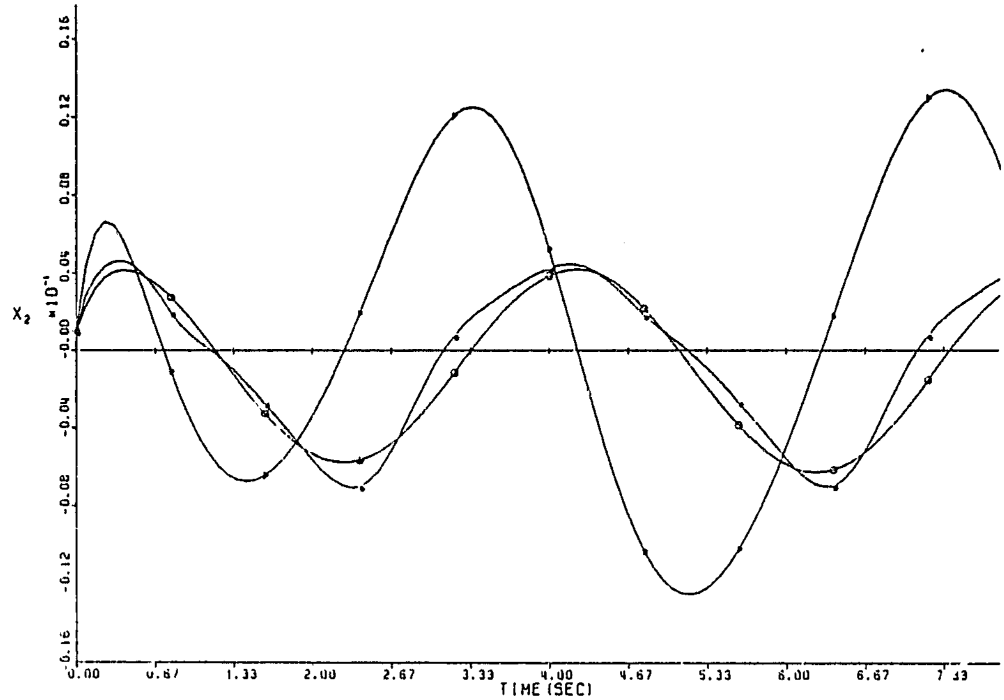

Figure 3 Table 2, Line 7

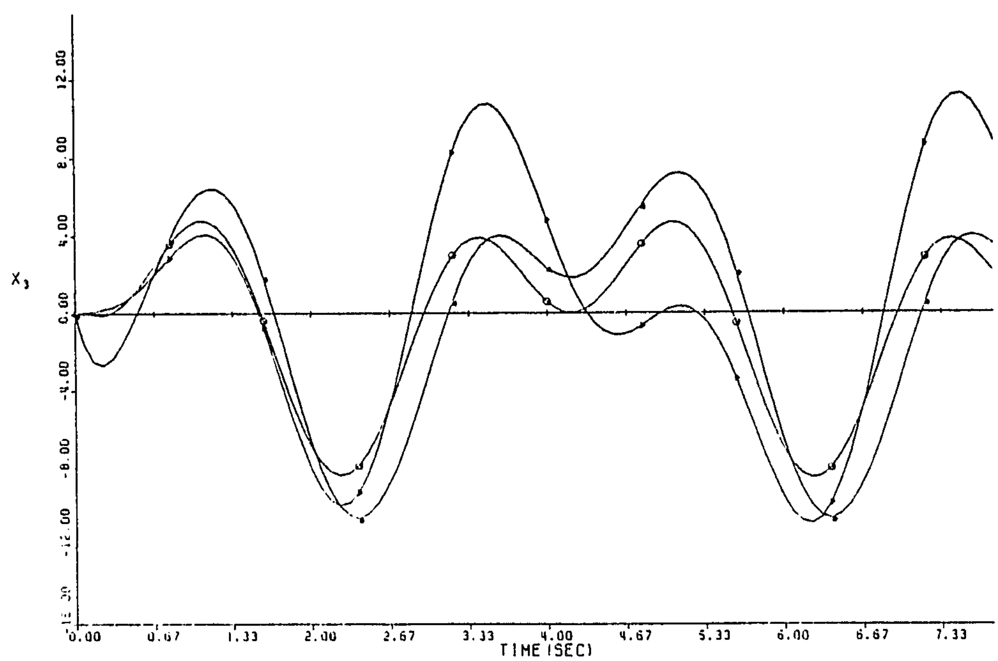

Figure 4 Table 2, Line 7

ORIGINAL PAGE IS
OF POOR QUALITY

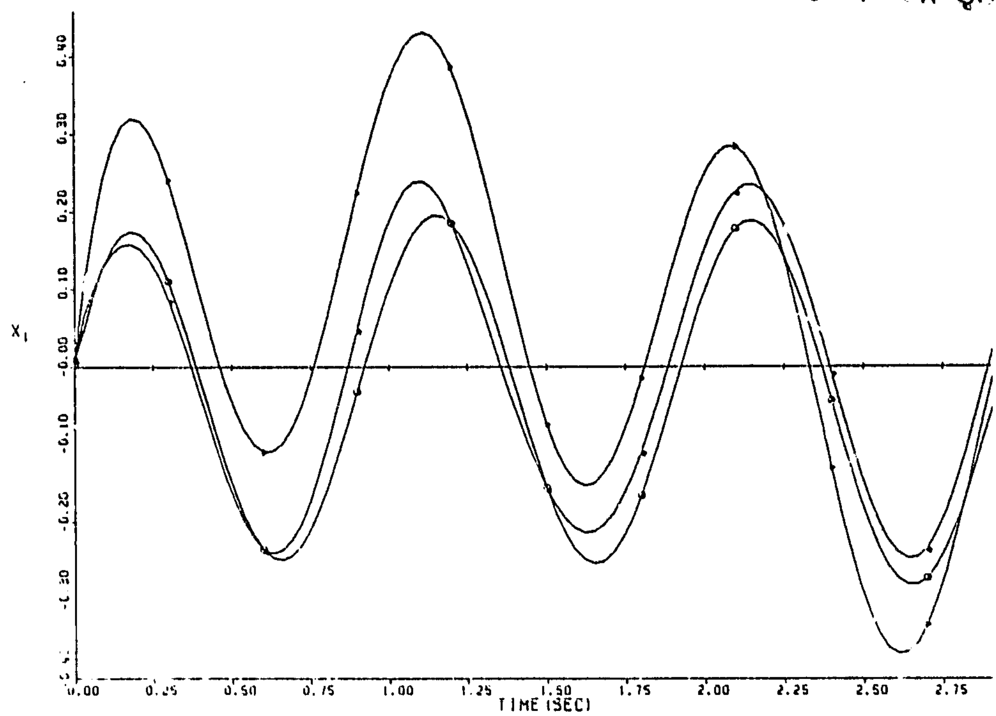

Figure 5 Table 2, Line 6

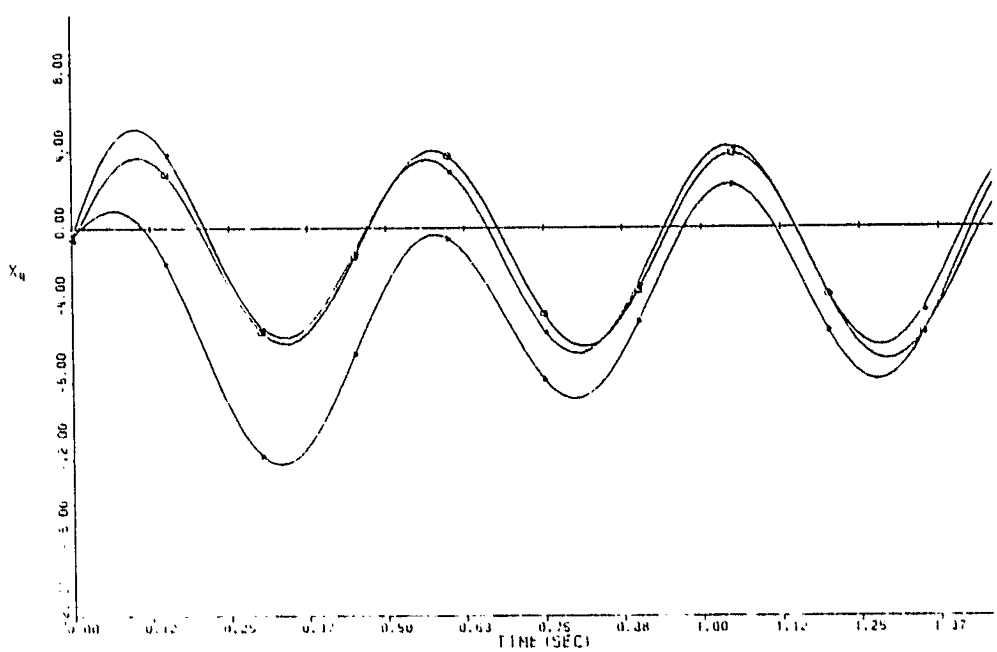

Figure 6 Off the Operating Line at 1/2%

ORIGINAL PAGE IS
OF POOR QUALITY

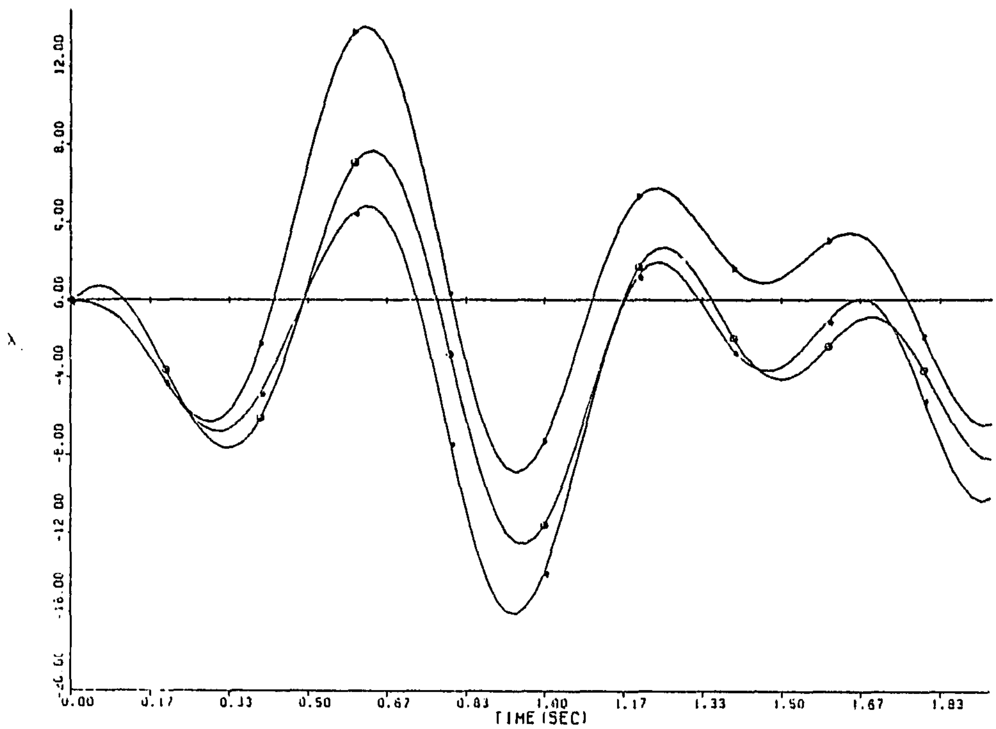

Figure 7 Table 2, Line 11

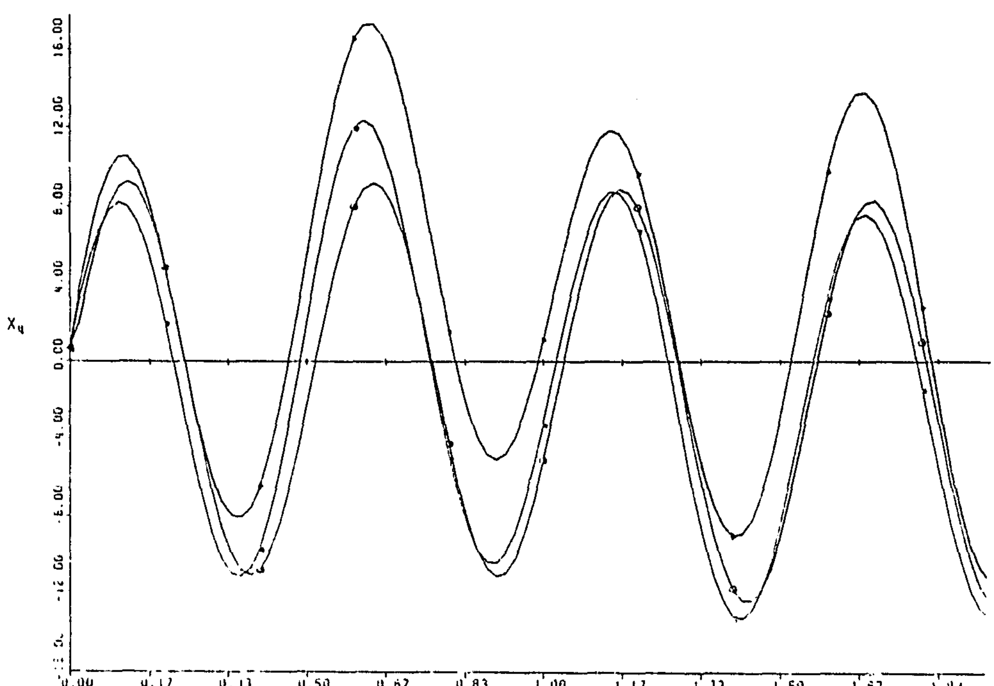

Figure 8 Table 2, Line 11

ORIGINAL PAGE IS
OF POOR QUALITY

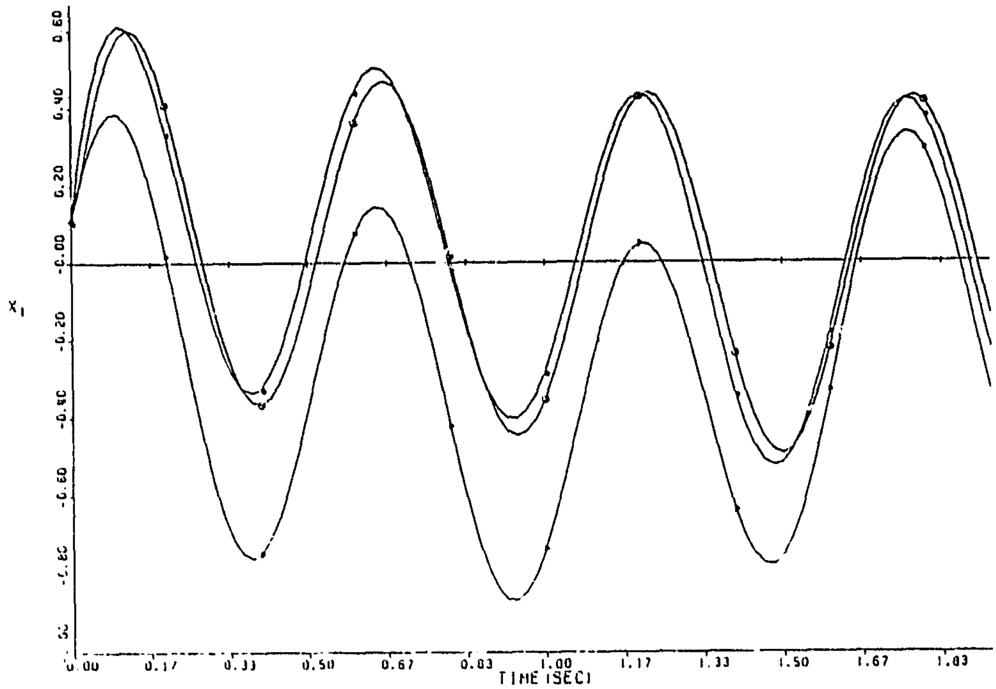

Figure 9 Table 2, Line 13

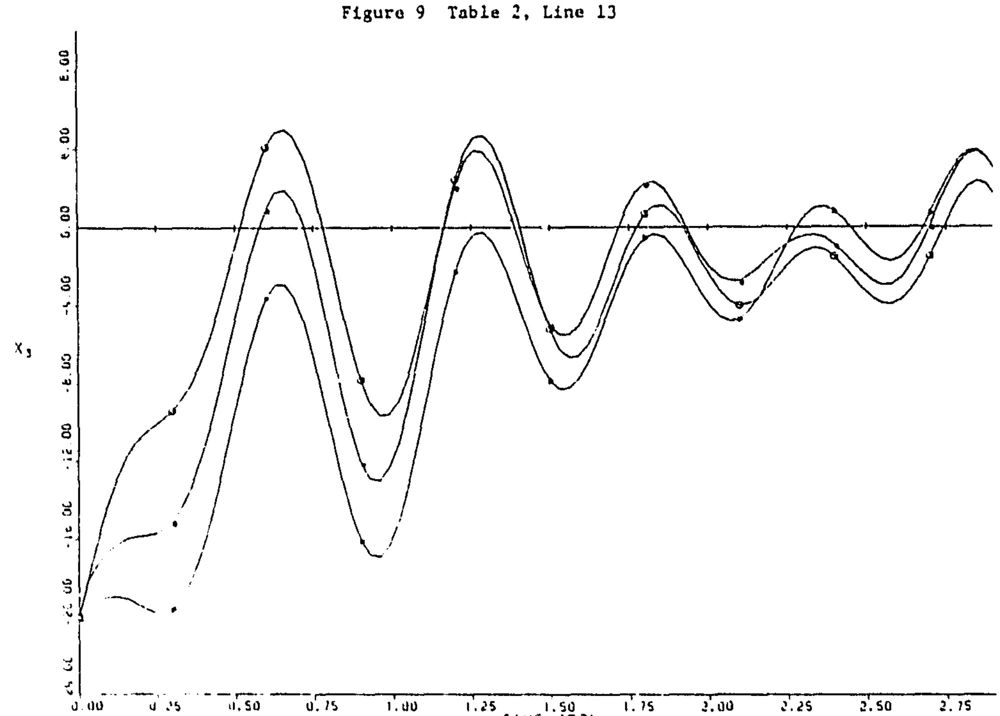

Figure 10 Table 2, Line 13

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX C

"Nonlinear Multivariable Design by Total Synthesis"

M.K. Sain
J.L. Peczkowski

Proceedings American Control Conference

Pages 252-260

June 1982

ORIGINAL PAGE IS
OF POOR QUALITY

NONLINEAR MULTIVARIABLE DESIGN BY TOTAL SYNTHESIS

Michael K. Sain
Electrical Engineering Department
University of Notre Dame
Notre Dame, Indiana 46556

Joseph L. Peczkowski
Energy Controls Division
The Bendix Corporation
South Bend, Indiana 46620

Abstract

In a recent publication [1], Bristol has presented an application theorist's view of process control design as it really exists and has challenged others to do likewise for areas within their own purview. This paper continues just such an effort [2,3,4] by the authors within the domain of nonlinear multivariable control of gas turbine engines. Under examination is the fundamental notion that linear controller descriptions, obtained from local actions of nonlinear objects, may be recombined to produce global nonlinear control action, with sufficient integrity to effect closed loop design. Total Synthesis refers to a top-down strategy of Nominal Design and Feedback Synthesis. This paper extends the study of the Nominal Design Problem (NDP) to nonlinear cases, and presents a new case study of robust feedback synthesis for gas turbine control design.

Introduction

The idea of describing families of curves by their tangents has a rich history in mathematics, in science, and in engineering. Consider, by way of example, the ubiquitous differential equation. More generally, the notions of manifold, tangent spaces, and geometry are very much a part of modern multivariable systems research.

Not surprisingly, the same notions permeate a great deal of control design in various applications. Intuitively, one linearizes a nonlinear dynamical system at a sequence of points along lines of operation considered desirable by the plant manufacturer. A suitably rich sequence of points can lead to a correspondingly valuable sequence of linear multivariable systems describing local gains and transient behavior of the plant along these operating lines. From such a sequence of systems one may construct a sequence of controllers which effect desirable local motions along the lines. Smooth global control is then a function of appropriately scheduled feedforward and feedback requests, as well as scheduling of local controller gains and dynamics which determine the approach to such requests.

Bristol [1] believes that experience and intuition are the crucial requisites for efficacious design, and that the best of design-flavored control theories can serve only as an introduction to the path followed by engineers with experience. Accordingly, Bristol also suggests that one should seek theories which extend one's intuition and which do not presuppose its replacement.

The local control theory employed by the authors in this paper was proposed [5] in 1979 by Peczkowski, Sain, and Leake, with just such a view in mind. Conceptually, the method is founded upon the idea of a Nominal Design Problem (NDP), which is independent of controller structure and which is intended as the first step in a top-down design procedure. A thorough discussion is given in [6]. This paper treats an extension of NDP to the nonlinear case. Completion of step one in NDP is followed by a second step, called the Feedback Synthesis Problem (FSP) [6]. A case study of this step may be found in [7], which also contains a full list of references. An alliance of NDP with FSP is called a Total Synthesis Problem (TSP). The case study following in this paper is part of a continuing assault of FSP for the nonlinear case, from the view of design practice in gas turbine engine control.

The section following provides mathematical preliminaries which precede a discussion of NDP for the nonlinear plant. Beyond that, nonlinear nominal design is defined and characterized, after which the paper progresses to design of local controllers for a turbojet case, and the scheduling of these local controls into a global control.

Mathematical Preliminaries

In this section, we consider a bijection $b : S \to T$ from a set S onto a set T, with T admitting the structure of an F-vector space. As a result of the fact that b is bijective, each vector t in T can be represented uniquely in the manner $b(s)$ for an s in S; and each element s in S can be represented uniquely by $b^{-1}(t)$ for a t in T. Here, we have denoted the inverse of b in the usual way, $b^{-1} : T \to S$.

The commutative group structure $(T, +, 0)$ on our F-vector space T can be used, together with b, to induce a commutative group structure $(S, \square, e)$ on the set S. The first step in this construction is to define the binary operation $\square : S \times S \to S$. We do this as follows. Let $(s_1, s_2) \in S \times S$; then

$$s_1 \square s_2 = b^{-1}(b(s_1) + b(s_2)) ,$$

where the binary operation $+$ in the right member is that on $T \times T$ to T. Associativity of the new operation can be demonstrated. Indeed,

$$(s_1 \square s_2) \square s_3 = b^{-1}(b \circ b^{-1}(bs_1 + bs_2) + bs_3)$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$= b^{-1}((bs_1 + bs_2) + bs_3)$$
$$= b^{-1}(bs_1 + (bs_2 + bs_3))$$
$$= b^{-1}(bs_1 + b \circ b^{-1}(bs_2 + bs_3))$$
$$= s_1 \square (s_2 \square s_3) \ .$$

The unit e can be chosen to be $b^{-1}(0)$, as is apparent from the calculation

$$s \square b^{-1}(0) = b^{-1}(bs + b \circ b^{-1}(0))$$
$$= b^{-1}(bs + 0)$$
$$= s \ .$$

For commutativity of the operation, we exhibit the steps

$$s_1 \square s_2 = b^{-1}(b(s_1) + b(s_2))$$
$$= b^{-1}(b(s_2) + b(s_1))$$
$$= s_2 \square s_1 \ .$$

Finally, for an element s in S, we define an additive inverse $\hat{s}$ in S to be $b^{-1}(-b(s))$, and verify it by

$$s \square \hat{s} = b^{-1}(bs + b \circ b^{-1}(-b(s)))$$
$$= b^{-1}(0)$$
$$= e \ ,$$

as desired. Accordingly, $(S, \square, b^{-1}(0))$ is a commutative group.

Next, we can use the scalar multiplication operation $F \times T \to T$ on the F-vector space T to induce a scalar multiplication $F \times S \to S$. To do this, we define the scalar multiple fs of s by f to be

$$fs = b^{-1}(fb(s)) \ ,$$

for a pair (f,s) in $F \times X$. Notice that

$$f(s_1 \square s_2) = b^{-1}(fb(s_1 \square s_2))$$
$$= b^{-1}(fb \circ b^{-1}(bs_1 + bs_2))$$
$$= b^{-1}(fbs_1 + fbs_2)$$
$$= b^{-1}(b \circ b^{-1}fbs_1 + b \circ b^{-1}fbs_2)$$
$$= b^{-1}(b(fs_1) + b(fs_2))$$
$$= (fs_1) \square (fs_2) \ .$$

Moreover, we can also see that

$$(f_1 + f_2)s = b^{-1}((f_1 + f_2)b(s))$$
$$= b^{-1}(f_1b(s) + f_2b(s))$$
$$= b^{-1}(b \circ b^{-1}f_1bs + b \circ b^{-1}f_2bs)$$
$$= b^{-1}(b(f_1s) + b(f_2s))$$
$$= (f_1s) \square (f_2s) \ .$$

Next, observe the property

$$(f_1f_2)s = b^{-1}((f_1f_2)bs)$$
$$= b^{-1}(f_1(f_2bs))$$
$$= b^{-1}(f_1(b \circ b^{-1}f_2bs))$$
$$= b^{-1}(f_1b(f_2s))$$
$$= f_1(f_2s) \ .$$

Finally,

$$1s = b^{-1}(1b(s))$$
$$= b^{-1}(b(s))$$
$$= s \ .$$

Thus, $(S, \square, b^{-1}(0))$ has been developed into an F-vector space S. We summarize this fact as a theorem.

<u>Theorem 1.</u>

Let $b : S \to T$ be a bijection onto the F-vector space $(T, +, 0)$. Then $(S, \square, b^{-1}(0))$ is also an F-vector space, with addition

$$s_1 \square s_2 = b^{-1}(b(s_1) + b(s_2)) \ ,$$

with additive inverse

$$b^{-1}(-b(s))$$

for a vector s, and with scalar multiplication

$$fs = b^{-1}(fb(s)) \ .$$

<u>Remark</u>

Frequently, S may be given as an F-vector space on a commutative group $(S, +, 0)$. Though the set S is common to these structures, the binary operations $+$ and $\square$ are distinct, as well as the units 0 and $b^{-1}(0)$ and the scalar multiplications.

Relative to the induced space, the bijection and its inverse assume desirable properties.

<u>Corollary 2.</u>

Regarded as a function

$$b : (S, \square, b^{-1}(0)) \to (T, +, 0) \ ,$$

the bijection b is a morphism of F-vector spaces, as is its inverse

$$b^{-1} : (T, +, 0) \to (S, \square, b^{-1}(0)) \ .$$

Proof: We have only to examine the defining constructions

ORIGINAL PAGE IS OF POOR QUALITY

$$b(s_1 \square s_2) = b \circ b^{-1}(b(s_1) + b(s_2))$$
$$= b(s_1) + b(s_2) ;$$
$$b(fs) = b \circ b^{-1}(fb(s))$$
$$= fb(s) ;$$

moreover, for each t_i in T, we have a unique s_i in S such that $t_i = b(s_i)$; and so we have also

$$b^{-1}(t_1 + t_2) = b^{-1}(b(s_1) + b(s_2))$$
$$= b^{-1} \circ b(s_1 \square s_2)$$
$$= s_1 \square s_2$$
$$= (b^{-1}(t_1)) \square (b^{-1}(t_2)) ;$$
$$b^{-1}(ft) = b^{-1}(fb(s))$$
$$= fs$$
$$= fb^{-1}(t) .$$

Remark

If R is a ring with identity, then all the discussions above generalize to R-modules.

Next, denote by S^R the set of all functions from a set R to the F-vector space $(S, \square, e)$. Under pointwise conventions,

$$(g_1 \square g_2)(r) = (g_1(r)) \square (g_2(r)) ,$$
$$(fg)(r) = fg(r) ,$$

S^R becomes an F-vector space also [8], as does T^R under the corresponding operations induced from T.

The following section defines the NDP for nonlinear plants, and uses the properties above to characterize its structure.

Nonlinear NDP

The concept of a nonlinear NDP was outlined in [3] by the authors for functions on commutative groups. Here we extend the idea. Let R, U, and Y denote F-vector spaces of requests to the system, controls to the plant, and responses from the plant. It is useful to visualize these, for example, as function spaces, predicated perhaps on time sets. Let $p : U \to Y$ denote the plant. If the feedback action of the controller is well defined, then there will be a function $m : R \to U$ generating control actions from requests and a function $t : R \to Y$ describing plant responses to requests. These three functions must then be related by the equation

$$t = p \circ m ,$$

which is presented as a commutative diagram in Figure 1. The nonlinear Nominal Design Problem is to find pairs (m,t) in (U^R, Y^R) such that the diagram of Figure 1 commutes. As usual, we point out

$$\begin{tikzcd} & R \arrow[dl, dashed, "m"'] \arrow[dr, dashed, "t"] & \\ U \arrow[rr, "p"'] & & Y \end{tikzcd}$$

Figure 1. Nonlinear NDP.

that NDP is not a model matching problem, in which t would also be given and in which only m in Figure 1 would appear on a dashed line.

Now consider a pair (m_i, t_i) of solutions to NDP. Characterization of the set $\{(m_\alpha, t_\alpha)\}$ of all solutions to NDP is severely hindered by the fact that

$$p \circ (m_1 + m_2) \neq (p \circ m_1) + (p \circ m_2) .$$

With the ideas of the section preceding, however, this difficulty can be addressed.

Let $(Y, +, 0)$ denote the F-vector space of plant responses, and let $(R, +, 0)$ denote the given F-vector space of requests. If p is a bijection, we can develop on U the F-vector space structure $(U, \square, p^{-1}(0))$ of Theorem 1. Relative to this structure, p and $p^{-1} : Y \to U$ become isomorphisms of F-vector spaces. Moreover, $(U^R, \square, e_{U^R})$ and $(T^R, +, 0)$ become F-vector spaces, with

$$e_{U^R}(r) = p^{-1}(0)$$

for all r in R; and $0(r) = 0$ in T for all r in R. We can then define the F-linear map $F : U^R \times T^R \to T^R$ by setting the action

$$F(m,t) = p \circ m - t .$$

This leads to the following result.

Theorem 3.

Let $p : U \to Y$ be a bijection onto the F-vector space $(Y, +, 0)$. Then a pair (m,t) is a solution to the nonlinear Nominal Design Problem if and only if

$$(m,t) \in \text{Ker } F .$$

Proof: The assertion is immediate. It may be worthwhile, however, to point out the F-vector product group structure on $U^R \times T^R$ defined by the operation

$$(m_1,t_1) * (m_2,t_2) = (m_1 \square m_2,\ t_1 + t_2) .$$

ORIGINAL PAGE IS
OF POOR QUALITY

<u>Remark</u>

If one wished, he or she could assume vector space structure on U, and define a special binary operation $\square$ on $T \times T$ by $p(p^{-1}(t_1) + p^{-1}(t_2))$.

<u>Remark</u>

An Inverse nonlinear NDP, denoted INDF, can be defined from the equation

$$m = p^{-1} \circ t .$$

<u>Remark</u>

Suppose that p were only surjective. It follows that p induces natural equivalence classes on U; and a projection $\pi : U \to U/\equiv$ can be defined. Then one has the universal factorization

$$p = \bar{p} \circ \pi$$

for a unique $\bar{p} : U/\equiv \to \underline{Y}$, which is a bijection. The structure $(U/\equiv, \square, \bar{p}^{-1}(0))$ can be developed, and NDP pursued again. Only equivalence classes of controls are determined.

The existence of plant inverses is of the first importance both in theory and in application design. In the next section, we examine briefly the turbojet engine model which will be used for our case study.

<u>Nonlinear Turbojet Model</u>

A nonlinear model of a simple turbojet engine is shown in Figure 2. It is representative, on a small scale, of the kind of nonlinear plant with which designers of turbine engines and turbine controls deal currently in practice. In essence, it is a computer simulation, typically constructed by engine manufacturers and provided to control manufacturers. The nonlinear turbojet model consists of three integrators, nine nonlinear functions, including five bi-variant functions, nine multipliers and dividers, and nine summing junctions. The model describes nonlinear dynamical and steady state relationships between three inputs: fuel flow, W_f, exhaust nozzle area, A_j, and turbine vane position, B, and six outputs: engine speed, N, turbine temperature, T4, engine thrust, FN, tailpipe pressure, P5, and two other outputs. We propose to think of the nonlinear simulation model as a nonlinear function p from a real vector space of control functions of time to a real vector space of plant response functions of time.

Locally, with appropriate technical assumptions, the nonlinear plant function can be approximated by a linear map $P : U \to Y$, in the neighborhood of a pair $(\bar{u},\bar{y})$ in the relation defined by p. When the plant function p is a linear map P, the transformation

$$P^{-1}(P(u_1) + P(u_2)) = u_1 + u_2 ,$$

for the usual vector space structure $(U, +, 0)$. Locally, then, the operation $\square : U \times U \to U$ can be replaced by $+ : U \times U \to U$.

Suppose next that the plant has an internal representation

$$\dot{x} = f(x,u) \quad , \quad y = g(x,u) ,$$

with appropriate smoothness conditions associated with the functions $f : X \times U \to X$ and $g : X \times U \to Y$. Let $\bar{x}$ be such that

$$f(\bar{x},\bar{u}) = 0 ,$$

and define

$$\Delta y = y - g(\bar{x},\bar{u}) \quad , \quad \Delta u = u - \bar{u} .$$

Then p may be assumed to have a local representation given by an impulse response operator

$$C\, e^{At} B + D$$

or by its transform, say $P(s)$, in the usual way.

The idea is to use these $P(s)$ to determine corresponding local descriptions of the parts of the controller, and then to schedule these parts together into a global whole.

<u>Remark</u>

In addition to the case study which follows, an accompanying paper [9] discusses some additional conceptual issues associated with such schedules.

For the following case study, three outputs have been selected for control: engine speed, N, turbine temperature, T4, and engine thrust, FN. The nonlinear engine model was identified locally at five conditions corresponding to 70%, 80%, 90%, 100%, and 110% speed levels. The engine input is given by

$$u = (W_f, A_j, B) ,$$

and the selected engine response vector is

$$y = (N, T4, FN) .$$

By way of illustration, at 100% speed conditions, the plant transfer function $P(s)$ and its inverse were found to be:

$$P(s) = \frac{\begin{bmatrix} 5.4(.01s+1) & 56.1(.01s+1) & -2704(.01s+1) \\ .13(1.5s+1) & -2.7(.50s+1)(.01s+1) & 336(.31s+1) \\ 2.4(.29s+1) & 68.3(.42s+1)(.01s+1) & 951(.01s+1) \end{bmatrix}}{(.50s+1)(.01s+1)}$$

$$P(s)^{-1} = \frac{\begin{bmatrix} .18(.23s+1)(.01s+1) & 1.7(.01s+1)(.007s+1) & -.083(.01s+1) \\ -.005(-.2s+1) & -.08(.01s+1) & .015(.01s+1) \\ -.0001(.74s+1)(.01s+1) & .0017(.01s+1)(.013s+1) & .00015(.01s+1) \end{bmatrix}}{(.009s+1)}$$

ORIGINAL PAGE IS
OF POOR QUALITY

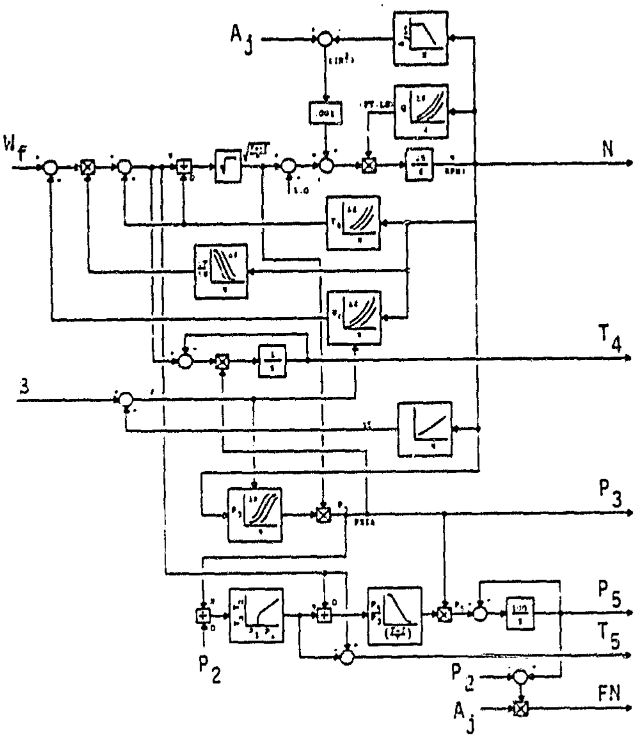

Figure 2. Nonlinear Turbojet Model.

Multivariable Design

A nonlinear multivariable design method, based on Total Synthesis ideas, is described and illustrated. Features of the design method include:

1. an input-output viewpoint;
2. design for desired response performance; control performance, and sensitivity;
3. a relatively general system structure;
4. a systematic way to synthesize the nonlinear controller.

Starting point for the design procedure is a nonlinear plant model or simulation such as the turbojet engine model shown in Figure 2. First, it is necessary to establish the desired steady state operating conditions of the plant and determine available plant inputs and plant outputs. Identification of the nonlinear plant along selected operating lines then can provide local plant dynamics and a set of plant transfer function matrices P(s) relating inputs and outputs.

Possibilities for control of plant outputs using available inputs can be checked by choosing subsets of square matrices of the plant transfer function matrix and determining the existence and condition of the corresponding plant inverse matrices. Existence of the plant inverse with good condition is necessary and vital to obtain reliable, independent control of selected outputs with available inputs [10-15].

Linear Design

A general linear system structure which combines TSP ideas with the idea of Comparison Sensitivity was presented and discussed in [4] and is shown in Figure 3. This system structure provides coordinated feedforward inputs u_r and loop command requests y_{rc}

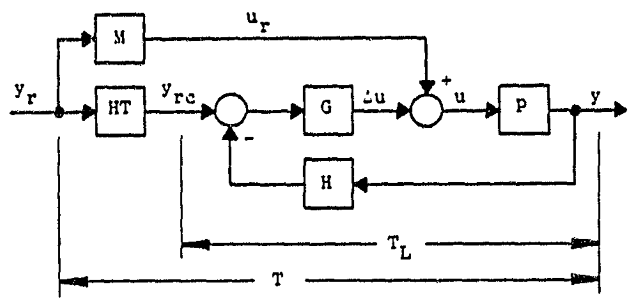

Figure 3. Robust Controller Structure.

to a closed loop control system. The feedforward elements coordinate request commands y_r; the closed loop assures steady state tracking and robustness of the outputs, y. The desired overall system response is designated by T. The chosen response of the loop is denoted by T_L.

Important controller elements of the structure are G, H, and M, which must be designed in an acceptable way so as to produce T and T_L within specifications. It turns out that three key equations govern local design:

$$M = P^{-1}T \qquad \text{(I)}$$

$$G = P^{-1}S_L^{-1}T_L \qquad \text{(II)}$$

$$H = T_L^{-1}(1-S_L) \qquad \text{(III)}$$

Equation (I) is called the synthesis equation. It is used to display all admissible responses (T,M) and (T_L,M_L). Equations (II) and (III) are design equations for the forward dynamics G and the feedback dynamics H, respectively. Note that all control dynamics are defined by selection of M, T, T_L and the comparison sensitivity S_L. The sensitivity S_L is defined [16,4] by $(1 + PGH)^{-1}$. These equations provide a basis to design linear control systems directly by specifying local response and sensitivity performance.

Nonlinear Design

As observed in the Introduction, the idea of describing families of curves by their tangents has a rich history in mathematics, in science, and in engineering. The method of phase plane portraits was already well developed more than two decades ago [17]. In more modern terms, we say today that

ORIGINAL PAGE IS
OF POOR QUALITY

state space descriptions, by ordinary differential equations, coincide with vector fields on manifolds [18]. Solutions on the manifold are curves tangent to the vectors of the field.

What about nonlinear design? The fundamental notion used in this paper is that linear descriptions, obtained from local actions of nonlinear objects, may be combined to produce nonlinear action, with sufficient integrity to effect closed loop control. For example, if a set of local designs has given rise to a family $\{G(s)\}$ of forward dynamics in the loop, then the goal is to link members of the family together so as to produce a nonlinear function g, that is $\{G\} \rightarrow g$. Now each G may be regarded as giving a local approximation to a part of the vector field. Under reasonable conditions of smoothness, and with enough members in the family, a careful linking could indeed lead to useful g, over regions of interest.

To extend this notion, one can consider choosing, along an operating line, sets of desired system responses $\{M,T\}$, loop responses $\{M_L, T_L\}$, and sensitivities $\{S_L\}$. From these performance choices, sets of controller matrices $\{M\}$, $\{G\}$, $\{H\}$, and $\{HT\}$ can be generated via equations (I), (II), and (III). All of the linear sets may be linked and scheduled as a function of plant condition to form nonlinear control elements. Thus $\{T\} \rightarrow t$, $\{M\} \rightarrow m$, $\{G\} \rightarrow g$, $\{H\} \rightarrow h$, $\{HT\} \rightarrow h \circ t$.

Desired steady state operating schedules, transient control means and protection limits are also needed to provide other practical and functional features for a nonlinear turbojet engine control system. These features transform the linear system structure in Figure 3 to the nonlinear system structure shown below, in Figure 4.

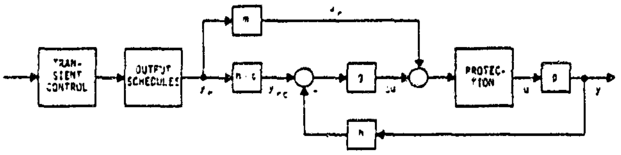

Figure 4. Nonlinear Control System.

The structure embodies key relationships of the Total Synthesis viewpoint and provides other basic features needed for full range, nonlinear control. It is used in the design examples which follow.

Design Examples

In this section we illustrate the foregoing synthesis ideas by designing a control system for the nonlinear turbojet engine described in Figure 2. Recall that the turbojet has three inputs: fuel flow, W_f, nozzle area, A_j, and turbine vane angle B; therefore, three outputs: engine speed, N, turbine temperature, T4, and engine thrust, FN were selected for control. We want to execute designs to achieve specific, beneficial output response strategies and show the effect that sensitivity specifications have in resisting plant parameter variations. Results are illustrated by time response traces for small step commands and by full range acceleration and deceleration transients.

Performance Specifications

Design a multivariable control system for full range acceleration and deceleration capability along the steady state schedules so that complete transients are completed in less than three seconds. Small signal responses of the system are desired to produce: 1) fast thrust response; 2) smooth, gentle temperature response; and 3) convenient speed response. All should take place without overshoot and without steady state error.

These requirements translate into the following kinds of response and sensitivity specifications:

1. Track output schedules with zero steady state error.
2. Accelerate or decelerate from 70% to 100% speed levels in less than 3 seconds.
3. Local System Responses (T) - Decoupled
   - Speed - .5 second lag @ 70% speed
   - - .2 second lag @ 100% speed
   - Temp. - .5 second lag - constant
   - Thrust - .2 second lag - constant.
4. Local Closed Loop Response (T_L) - Decoupled
   .2 second lag constant for all outputs.
5. Local Sensitivity (S_L)
   a) Unity feedback: $S_L = (I-T_L)$;
   b) Ten times better than unity feedback

The following response and sensitivity matrices at 100% speed condition were obtained:

$$T = \text{diag}\left(\frac{1}{.2s+1}, \frac{1}{.5s+1}, \frac{1}{.2s+1}\right)$$

$$T_L = \text{diag}\left(\frac{1}{.2s+1}, \frac{1}{.2s+1}, \frac{1}{.2s+1}\right)$$

$$S_L = \text{diag}\left(\frac{.2s}{.2s+1}, \frac{.2s}{.2s+1}, \frac{.2s}{.2s+1}\right) \quad (H=I)$$

$$S_L = \text{diag}\left(\frac{.02s}{.02s+1}, \frac{.02s}{.02s+1}, \frac{.02s}{.02s+1}\right)$$

The forward control dynamics $G(s)$ at 100% speed condition with unity feedback are:

$$G(s) = \frac{\begin{bmatrix} .90(.23s+1) & 8.5(.01s+1) & -.42(.01s+1) \\ -.025(-.2s+1) & -.40(.01s+1) & .075(.01s+1) \\ -.0005(.74s+1) & .0085(.013s+1) & .0008(.01s+1) \end{bmatrix}}{s(.01s+1)}$$

Controller elements M, G, H and HT were calculated at five engine speed conditions. These sets were scheduled as a function of engine speed

ORIGINAL PAGE IS
OF POOR QUALITY

to form nonlinear control system elements m, g, h and h • t. For example, the form of the nonlinear controller g so constructed is shown in Figure 5 below.

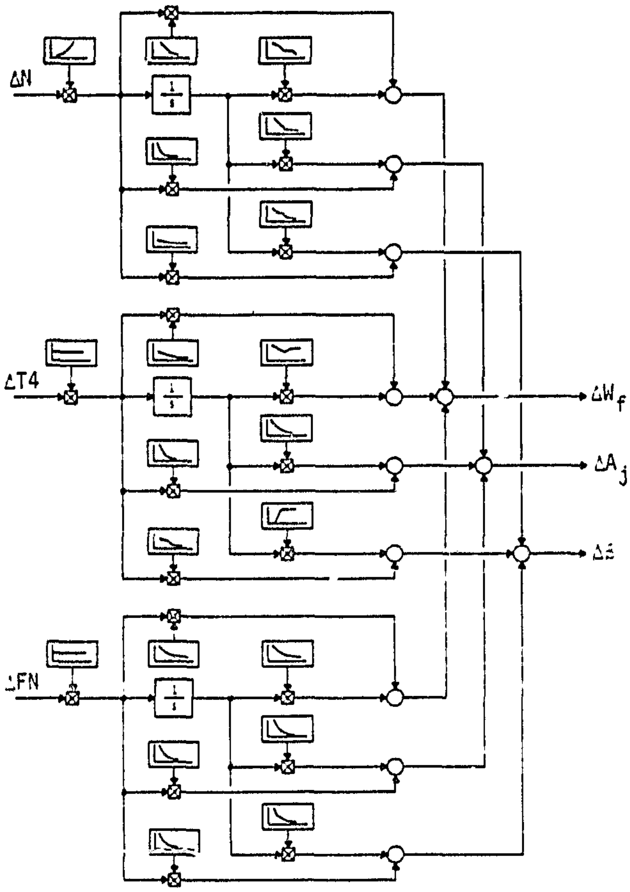

Figure 5. Nonlinear g.

Simulation Results

Small step transients of the nominal engine with sensitivity feedback system are shown in Figure 6. The output responses verify desired small signal performance: thrust response is fast (.2 second lag); temperature response is smooth (.5 second lag), and speed response varies from .5 second lag at 70% to .2 second lag at 100% speed condition. Corresponding input responses are shown in Figure 7.

Full range acceleration and deceleration transients of the nominal engine with sensitivity feedback system are shown in Figure 8. The outputs track the requests without overshoot and the transient time is less than three seconds. Corresponding input responses are shown in Figure 9. All inputs are within desired limits.

To show the effect of sensitivity specification on plant parameter variations, the time constants of the engine affecting speed, temperature

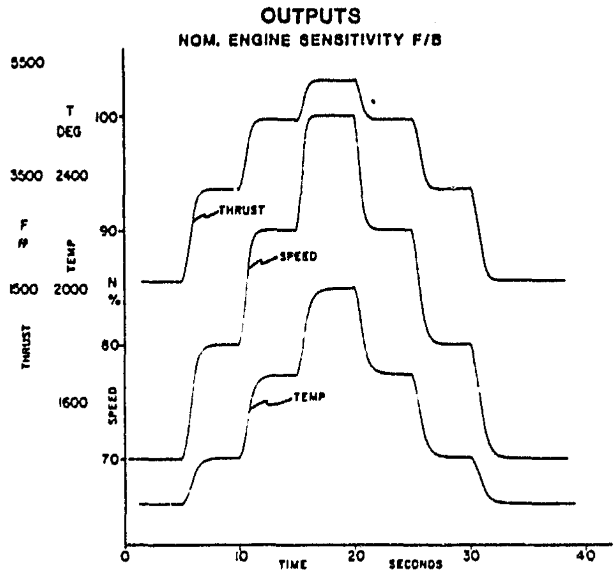

Figure 6. Outputs.
Nominal Engine; Sensitivity F/B.

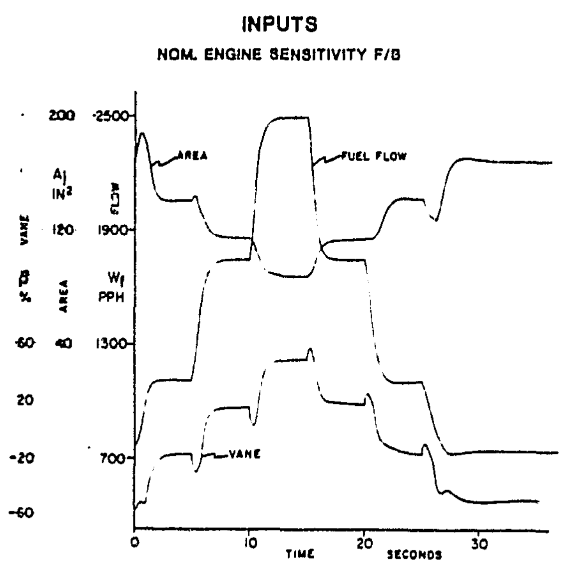

Figure 7. Inputs.
Nominal Engine; Sensitivity F/B.

and tail pipe pressure response were all doubled. This produced a nonnominal engine.

Small step responses of a nominal and nonnominal engine with unity feedback control are pictured in Figure 10. Deviations from desired output responses are caused by the engine parameter variations.

Imposing a sensitivity specification which is effectively ten times better than the unity feedback design produced feedback dynamics H = diag ((.2s+1)/(.02s+1)). Step responses for the sensi-

ORIGINAL PAGE IS
OF POOR QUALITY

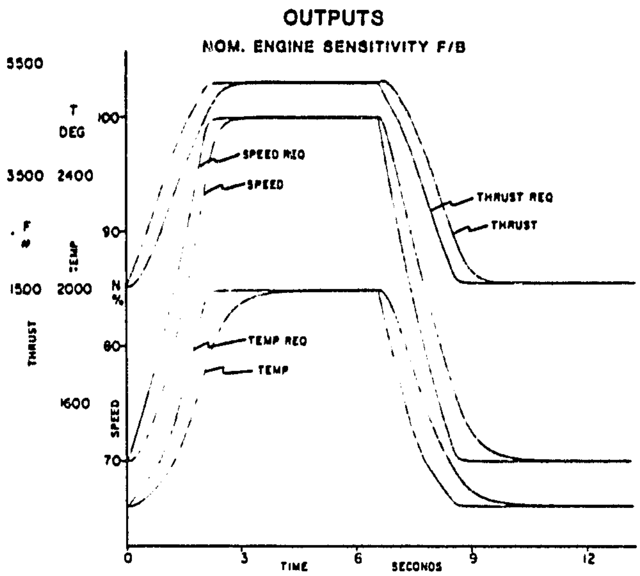

Figure 8. Outputs.
Nominal Engine ; Sensitivity F/B.

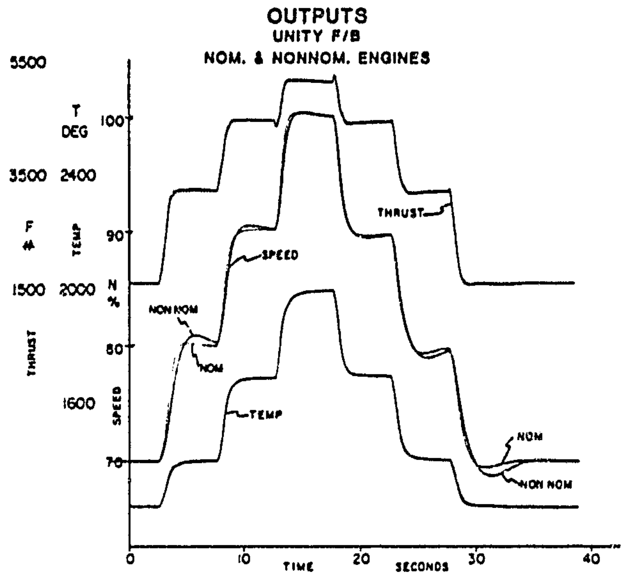

Figure 10. Outputs; Unity F/B.
Nominal and Nonnominal Engines.

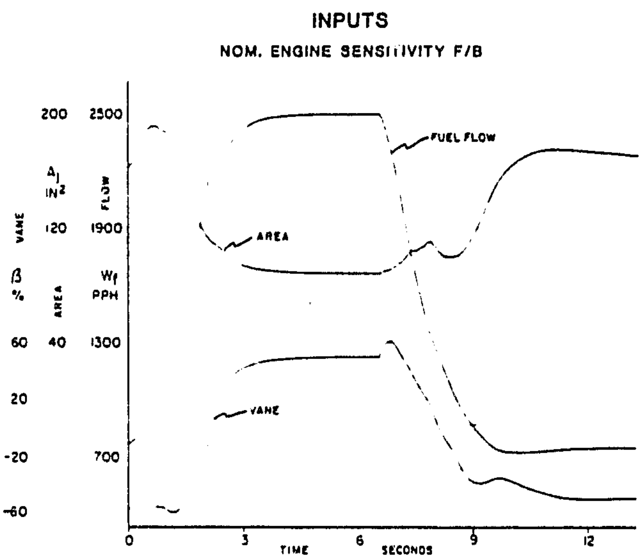

Figure 9. Inputs.
Nominal Engine; Sensitivity F/B.

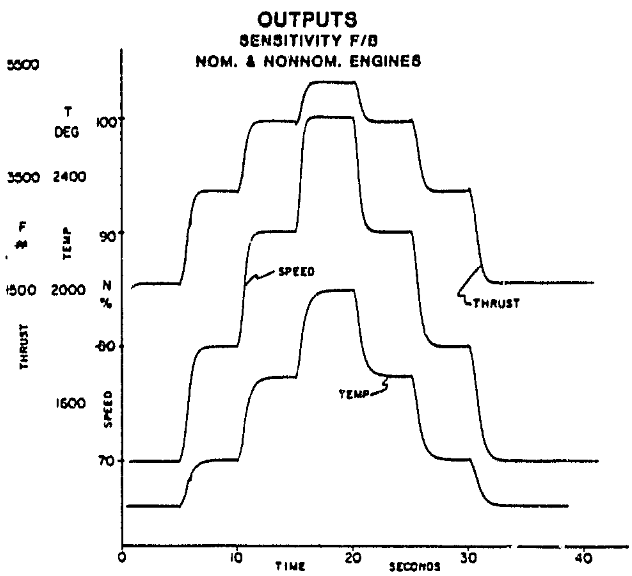

Figure 11. Outputs; Sensitivity F/B.
Nominal and Nonnominal Engines.

tivity feedback design with both the nominal and nonnominal engines are shown in Figure 11. Deviations due to engine parameter variations are virtually eliminated.

Full range acceleration and deceleration transients of the sensitivity feedback controller with both nominal and nonnominal engines are shown in Figure 12. The sensitivity design feedback controller maintains full range output transients essentially at nominal conditions, successfully handling plant parameter variations.

Summary Remarks

A nonlinear control synthesis method based on TSP viewpoint was discussed and illustrated. A three input/three output turbojet engine example demonstrated a feasibility to achieve desired system response and sensitivity specifications.

A concept of the nonlinear Nominal Design Problem (NDP) was presented and discussed, extending and building on earlier Total Synthesis Problem (TSP) theory and ideas. Additive structure was ob-

ORIGINAL PAGE IS
OF POOR QUALITY

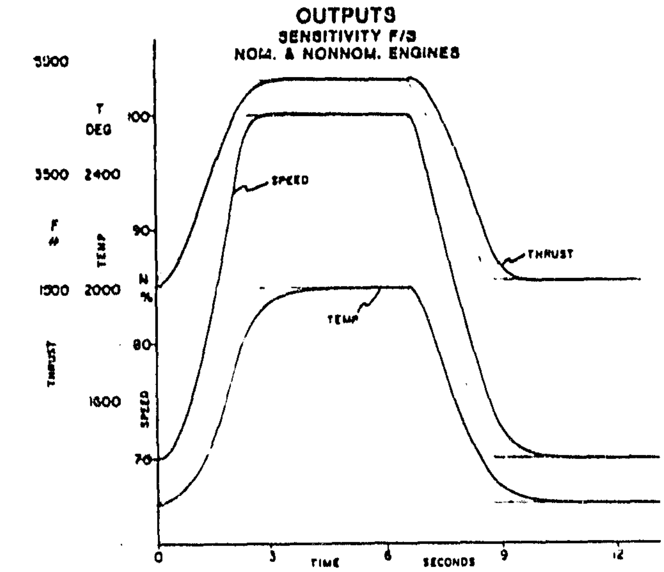

Figure 12. Outputs; Sensitivity F/B. Nominal and Nonnominal Engines.

tained by a process of inducing a special binary operation on the control input space. Though not a new mathematical idea [18], this concept seems to fit constructively into current design developments in nonlinear control.

Research to develop nonlinear control synthesis methods is needed. It is felt that the input-output TSP viewpoint offers possibilities to develop useful, systematic and straightforward methods for nonlinear multivariable control synthesis.

Acknowledgments

The support and extraordinary contribution of Mr. Ben Jacobs in simulation and verification of the design on our AD-10 hybrid computer is gratefully acknowledged. The assistance of Betty Raven in identification of the engine model is also gratefully acknowledged. Ben and Betty are both with the Bendix Energy Controls Division.

This work also benefited in part from support extended to the first author by the National Aeronautics and Space Administration under Grant NSG-3048.

References

1. E. Bristol, "Process Control: An Application Theorist's View of Control", IEEE Control Systems Magazine, Vol. 2, No. 1, pp. 3-8, March 1982.
2. J.L. Peczkowski and S.A. Stopher, "Nonlinear Multivariable Synthesis with Transfer Functions", Proc. 1980 Joint Automatic Control Conf., Vol. 1, Pt. WA8-D.
3. J.L. Peczkowski and M.K. Sain, "Scheduled Nonlinear Control Design for a Turbojet Engine", Proc. 1981 IEEE Int. Symp. on Circuits and Systems, pp. 248-251, April 1981.
4. M.K. Sain and J.L. Peczkowski, "An Approach to Robust Nonlinear Control Design", Proc. 1981 Joint Automatic Control Conf., Vol. 2, Pt. FA-3D.
5. J.L. Peczkowski, M.K. Sain, and R.J. Leake, "Multivariable Synthesis with Inverses", Proc. 1979 Joint Automatic Control Conf., pp. 375-380.
6. M.K. Sain, B.F. Wyman, R.R. Gejji, P.J. Antsaklis, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part I: Nominal Design", Proc. 1981 Joint Automatic Control Conf., Paper WP-4A.
7. M.K. Sain, P.J. Antsaklis, B.F. Wyman, R.R. Gejji, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part II: Unity Feedback and the Design Morphism", Proc. 1981 IEEE Conf. on Decision and Control, pp. 875-884.
8. M.K. Sain, Introduction to Algebraic System Theory. New York: Academic Press, 1981.
9. M.K. Sain and S. Yurkovich, "Controller Scheduling: A Possible Algebraic Viewpoint", Proc. 1982 American Control Conference.
10. L.M. Silverman and H.J. Payne, "Input-Output Structure of Linear Systems with Application to the Decoupling Problem" SIAM J. on Control, Vol. 9, pp. 199-233, May 1971.
11. B.C. Moore and L.M. Silverman, "Exact Model Matching by State Feedback and Dynamic Compensation", IEEE Trans. on Automatic Control, Vol. AC-17, pp. 491-497, August 1972.
12. S.-H. Wang and E.J. Davison, "A Minimization Algorithm for the Design of Linear Multivariable Systems", IEEE Trans. on Automatic Control, Vol. AC-18, pp. 220-225, April 1973.
13. W.M. Wonham and A.S. Morse, "Decoupling and Pole Assignment in Linear Multivariable Systems: A Geometric Approach", SIAM J. on Control, Vol. 8, pp. 317-337, 1970.
14. Z.V. Rekasius, "Linear Multivariable Control: A Problem of Specification", Proc. Int. Forum on Alternatives for Linear Multivariable Control, National Engineering Consortium, pp. 101-110, October 1977.
15. B.F. Wyman and M.K. Sain, "The Zero Module and Essential Inverse Systems", IEEE Trans. on Circuits and Systems, Vol. CAS-28, pp. 112-126, February 1981.
16. J.B. Cruz, Jr. and W.R. Perkins, "A New Approach to the Sensitivity Problem in Multivariable Feedback Design", IEEE Trans. on Automatic Control, Vol. AC-9, pp. 216-223, July 1964.
17. G.J. Thaler and M.P. Pastel, Analysis and Design of Nonlinear Feedback Control Systems. New York: McGraw-Hill, 1962.
18. W.M. Boothby, An Introduction to Differentiable Manifolds and Riemannian Geometry. New York: Academic Press, 1975.

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX D

"Controller Scheduling: A Possible Algebraic Viewpoint"

M.K. Sain
S. Yurkovich

Proceedings American Control Conference

Pages 261-269

June 1982

ORIGINAL PAGE IS
OF POOR QUALITY

CONTROLLER SCHEDULING: A POSSIBLE ALGEBRAIC VIEWPOINT

Michael Sain and Stephen Yurkovich
Department of Electrical Engineering
University of Notre Dame
Notre Dame, Indiana 46556

Abstract

In the applications, one common way to design a control system for a nonlinear plant is to localize its behavior along lines of operation specified by the plant manufacturer, to develop linear multivariable controls for these localizations, and to schedule those controls with key plant variables which vary smoothly along operating lines. An important part of practical design lore, the art of controller scheduling has received little modern attention from the conceptual point of view. This paper describes four basic types of scheduling questions and outlines some of the theoretical issues associated with them. Schedules are considered in terms of state equations; however, some relations with the input/output description are discussed, together with an analysis of the effects on the overall configuration of approximations made to the individual subsystems.

Introduction

Bristol [1,2] has likened the process of control design to the use of idioms in a language. At least three types of idioms can be identified. First, there are idioms which have been with mankind for such a length of time that they seem universal to the human psyche. In some sense, feedback itself is an example of such an idiom, inasmuch as it may be traced at least back to ancient Arabian water clocks. Second, there are idioms which are the characteristic of certain authors. Several classic examples are the Nichols chart, the Bode plot, the Evans loci, and the Nyquist plot. And third, there are idioms which are typical of certain types of control applications. An example is that of gas turbine control systems [3].

Because of the idioms of type three, any application of control design has idiomatic features. In a sense, the task of the control designer is to blend the idioms of the application with universal idioms, with idioms of classical and modern authors, and with his or her own idioms, so as to produce a melodious and effective composition.

It goes without saying that some idioms do not play well together. In some areas of application, this may account for the famous theory/application gap.

One universal idiom is to attack the overall system design by breaking it down into manageable pieces. An important case of this type of thinking arises in the design of certain classes of nonlinear systems. Examples in point may be found in the area of gas turbine control. For discussion of some of the ideas involved, as well as additional references, see the companion paper [4]. In brief, the nonlinear engine is linearized locally along lines of operation agreed upon by the manufacturer and the control contractor. These linear multivariable localizations are used to develop a family of local controllers, which are then sewn together by scheduling control gains and dynamics with some engine variable, as for example speed, which varies smoothly along operating lines.

As pointed out by Bristol [1], the idioms have to blend together. In the case of scheduling, the methods used for design of the local, linear multivariable controllers have to be amenable to a common thread of smooth scheduling, else a global whole is not obtained, but only a sum of parts.

The goal of this paper is to examine in an introductory way certain of the conceptual questions associated with scheduling. In particular, we would like to determine something about the structures of common scheduled systems, their approximations, and how they are affected by interconnections one with the other.

What follows should be regarded as exploratory in nature. Though we will do some things in considerable detail, it nonetheless remains true that we will be answering only a portion of the questions which we raise.

Simple Example

Consider the elementary dynamical system

$$\dot{x} = -ax + bu \ . \qquad (1)$$

The transfer function associated with (1) is of course

$$\frac{b}{s+a} \ . \qquad (2)$$

Rewritten in terms of gain and time constant, (2) becomes

$$\frac{k}{\tau s+1} \ , \qquad (3)$$

where

$$k = b/a \quad , \quad \tau = 1/a \ . \qquad (4)$$

Suppose that we wanted to schedule the gain k as a function of the input u, say

ORIGINAL PAGE IS OF POOR QUALITY

$$k(u) = \alpha_1 + \beta_1 u + \gamma_1 u^2 . \qquad (5)$$

Then the scheduled system would look like

$$\dot{x} = -ax + a\alpha_1 u + a\beta_1 u^2 + a\gamma_1 u^3 . \qquad (6)$$

Alternatively, we might schedule the time constant τ as such a function, for example

$$\tau(u) = \alpha_2 + \beta_2 u + \gamma_2 u^2 , \qquad (7)$$

in which case we would have

$$\begin{aligned} a &= 1/(\alpha_2 + \beta_2 u + \gamma_2 u^2) \\ &= \alpha_2^{-1} - \beta_2 \alpha_2^{-2} u + \ldots \end{aligned} \qquad (8)$$

so that

$$\dot{x} = -\alpha_2^{-1} x + \beta_2 \alpha_2^{-2} ux + bu + \ldots \qquad (9)$$

on out to a denumerably infinite number of terms. Next suppose that we wanted to schedule the gain k or the time constant τ as a function not of u but of x, in the manner

$$k(x) = \alpha_3 + \beta_3 x + \gamma_3 x^2 , \qquad (10)$$

$$\tau(x) = \alpha_4 + \beta_4 x + \gamma_4 x^2 . \qquad (11)$$

Then the scheduled systems would be

$$\dot{x} = -ax + a\alpha_3 u + a\beta_3 xu + a\gamma_3 x^2 u, \qquad (12)$$

$$\dot{x} = -\alpha_4^{-1} x + \beta_4 \alpha_4^{-2} x^2 + bu + \ldots , \qquad (13)$$

again with a denumerably infinite number of terms.

Generally speaking, the polynomic scheduling concept tends to convert the system (1) into a system of the form

$$\dot{x} = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} r_{ij} x^i u^j . \qquad (14)$$

Indeed, if the original system (1) were of the more general form

$$\dot{x} = \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} a_{km} x^k u^m , \qquad (15)$$

and if the parameters were scheduled in an analogous way, such as

$$a_{km} = \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} a_{kmpq} x^p u^q , \qquad (16)$$

then (15) becomes

$$\dot{x} = \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} \sum_{p=0}^{\infty} \sum_{q=0}^{\infty} a_{kmpq} x^{k+p} u^{m+q} , \qquad (17)$$

which can be formally rearranged in the same form as (15). In broad terms, then, (15) is closed under formal power series scheduling.

Because of this closure feature, we find interest in systems of this type. The next section gives a brief, multivariable motivation.

Multivariable Motivation

When a nonlinear plant has been linearized in the neighborhood of a point on the desired operating line, the resulting, local, linear multivariable control problem must be resolved by some chosen procedure. Moreover, associated with this chosen method will be various related and compatible theoretical viewpoints. It is not the purpose of this paper to argue the merits of one or more of these theories. Instead, we wish to select a way of thinking which is comprehensive enough to embrace the thoughts of numerous approaches to such conceptualization. In this way, the scheduling classes which we introduce will, hopefully, be broad and representative of those which arise in a variety of schemes. Of course, not every class would be natural to every theory; and not every possible scheduling class can be encompassed by any one viewpoint. The idea, nonetheless, is to generate, as it were, some of the characteristic features encountered in scheduling.

For the above mentioned, illustrative, purposes, then, we select the Total Synthesis Problem (TSP) structure proposed by Peczkowski, Sain, and Leake [5] in 1979. In the TSP idea, both the command/output-response, represented say by a matrix $[T(s)]$, and the command/control-response, represented say by a matrix $[M(s)]$, are to be simultaneously synthesized subject to the constraint imposed by a plant, which could be represented by a matrix $[P(s)]$. Fundamental to TSP is the Nominal Design Equation (NDE)

$$[T(s)] = [P(s)][M(s)] ,$$

which must be satisfied no matter what controller structure might be selected. For a full discussion of NDE, see [6]. Once NDE is solved, one can make use of any of the multitudinous structural synthesis methods of modern theory or modern practice to develop a feedback controller. Feedback synthesis generates a number of interesting questions of theory. For examples of these and for additional references, see [7]. It also generates very important questions of practice. Examples of these have been given in [8,9,10]. In [8], the structure of Figure 1 was employed on a turbojet

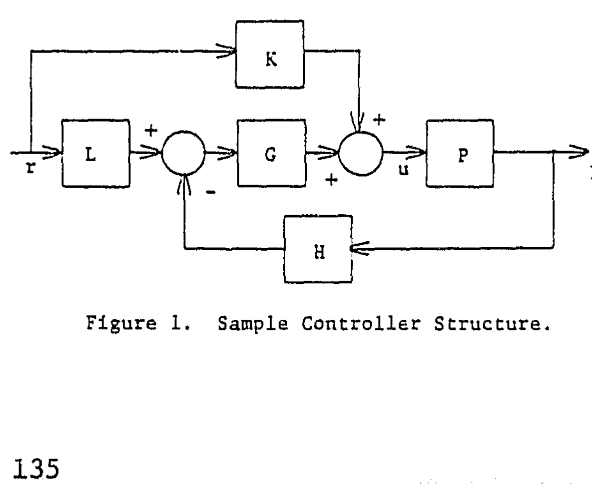

Figure 1. Sample Controller Structure.

ORIGINAL PAGE IS
OF POOR QUALITY

example. Subsequently, [9] and [10] considered particularizations of [8], especially insofar as L and K are concerned.

The structure [8] of Figure 1, however, is adequate for purposes of subsequent discussion. Let families $\{L,K,G,H\}$ have been determined by a technique of the reader's choice, in correspondence to a family $\{P\}$ developed along desired operating lines of the plant. From a more global point of view, the plant may be regarded as a general, not necessarily linear, function $p : U \to Y$ for appropriate control and response spaces U and Y. We are interested in considering four cases of scheduling:

(I) L and K as a function of r;
(II) L and K as a function of their output;
(III) G and H as a function of Lr;
(IV) G and H as a function of y.

Notice that, in I and II, we may as well choose either L or K, because the issues are the same.

Abstract Derivatives

As indicated in the Simple Example Section above, the idea of polynomic scheduling, of gains or time constants, suggests a state description in terms of series. Because we wish to use operator theoretic methods to some extent, it is convenient here to make a few introductory remarks about derivatives in such a context.

Let V and W be normed real vector spaces, with Z open in V. A function $f : Z \to W$ is differentiable at a point p in Z if there exists a continuous linear map $F : V \to W$ such that, for $(p+h)$ in Z and h in V,

$$\lim_{||h||\to 0} \frac{||f(p+h) - f(p) - Fh||}{||h||} = 0 .$$

If F exists, then it is unique and is called the derivative of f at p, and is denoted by

$$(Df)(p) : V \to W .$$

In case f is differentiable on Z, then we have a construction

$$Df : Z \to L(V,W) ,$$

where $L(V,W)$ denotes the real vector space of $\mathbb{R}$-linear maps $V \to W$. Higher order derivatives are defined in a recursive fashion,

$$(D^r f)(p) = (D(D^{r-1}f))(p) ,$$

with r a positive integer, provided that the indicated limit exists. For more discussion of these notions, the reader may wish to consult [11, 12,13].

An important connection exists between the calculus on normed vector spaces and the tensor algebra. Indeed,

$$D^2f(p) \in L(V,L(V,W)) ,$$
$$D^3f(p) \in L(V,L(V,L(V,W)))$$
$$\vdots \qquad\qquad \vdots$$

whenever the limits exist. Let us denote by

$$L(V_1,V_2,\ldots,V_n,W)$$

the real vector space of n-linear functions

$$V_1 \times V_2 \times \ldots \times V_n \to W ,$$

an n-linear function being one which is linear in its remaining argument whenever $(n-1)$ of its arguments are fixed. It can be shown that there exist isomorphisms

$$L(V_1,V_2,W) \to L(V_1,L(V_2,W)) ,$$
$$L(V_1,V_2,V_3,W) \to L(V_1,L(V_2,L(V_3,W))) ,$$
$$\vdots \qquad\qquad \vdots$$

so that $(D^r f)(p)$ can be regarded as an r-linear map $V^r \to W$, up to isomorphism. We suppress this isomorphism and think of $(D^r f)(p)$ as just such a map.

It is now straightforward to establish a connection with the tensor algebra, and we do so in the section following. The importance of the connection lies in its parametric possibilities: Every r-linear map can be composed from a linear map and a universal r-linear construction called tensor product. In a sense, the linear map embodies the parameters which are available for scheduling; and we pursue this view in a later section.

Tensor Algebra

In this section, we develop some of the structures with which we can subsequently discuss scheduling questions I-IV. Let V be a real vector space. For each integer r which is two or greater, let

$$(\otimes^r V, \otimes^r)$$

be a tensor product for r copies of V. The notion extends to 1 and 0 by the definitions

$$\otimes^1 V = V \quad , \quad \otimes^0 V = \mathbb{R} .$$

The sequence $\otimes^r V$, $r = 0,1,2,\ldots,$ can be developed into a biproduct, and the images of $\otimes^r V$ under insertion can be given the same notation. Then the tensorial powers $\otimes^n V$ can be developed into an associative algebra by defining the internal direct sum

$$\otimes V = \sum_{n=0}^{\infty} \otimes^n V ,$$

ORIGINAL PAGE IS
OF POOR QUALITY

and by equipping $\otimes V$ with the bilinear mapping $(\alpha,\beta) \to \alpha\beta$ for $\alpha, \beta, \alpha\beta \in \otimes V$ whose result is defined by

$$\alpha\beta = \sum_{n,m} \alpha_n \otimes \beta_m ,$$

where $\alpha = \sum_n \alpha_n$, $\beta = \sum_m \beta_m$ for $\alpha_n \in \otimes^n V$ and $\beta_m \in \otimes^m V$. With this multiplication, $\otimes V$ becomes the graded tensor algebra over V with elements $(\alpha_0, \alpha_1, \ldots)$, which are sequences of the tensors $\alpha_i \in \otimes^i V$, $i = 0,1,\ldots,$ and with unit element $(1,0,\ldots)$. We emphasize the fact that multiplication in the tensor algebra is not a tensor product.

Now let $\otimes V$ and $\otimes W$ be tensor algebras as defined above, over V and W respectively. For every pair $n,m \geq 1$, let $\otimes^n_m(V,W)$ be a tensor product of $\otimes^n V$ and $\otimes^m W$, that is,

$$\otimes^n_m(V,W) = (\otimes^n V) \otimes (\otimes^m W) .$$

We set $\otimes^n_0(V,W) = \otimes^n V$ and $\otimes^0_m(V,W) = \otimes^m W$. In a manner similar to that preceding,

$$\otimes^n_m(V,W) \quad , \quad n = 0,1,2,\ldots, \ m = 0,1,2,\ldots$$

can also be developed into a biproduct; and the images of each of these spaces under natural insertion into the biproduct can again be given the same symbolic representation. Again, then, we construct the internal direct sum

$$\otimes(V,W) = \sum_{n,m\geq 0}^{\infty} \otimes^n_m(V,W)$$

with

$$\otimes(V,W) = \sum_{k=0}^{\infty} [\sum_{n+m=k} \otimes^n_m(V,W)]$$

functioning as the induced gradation on $\otimes(V,W)$.

Now consider two spaces $\otimes^n_m(V,W)$ and $\otimes^r_s(V,W)$. There exists a unique bilinear mapping

$$\mu : \otimes^n_m(V,W) \times \otimes^r_s(V,W) \to \otimes^{n+r}_{m+s}(V,W)$$

with action

$$\mu(\alpha_n \otimes \beta_m, \alpha_r \otimes \beta_s) = (\alpha_n \otimes \alpha_r) \otimes (\beta_m \otimes \beta_s) ,$$

where $\alpha_n \in \otimes^n V$, $\alpha_r \in \otimes^r V$, $\beta_m \in \otimes^m W$, $\beta_s \in \otimes^s W$. The pair $(\otimes^{n+r}_{m+s}(V,W),\mu)$ is a tensor product, or

$$\otimes^{n+r}_{m+s}(V,W) = \otimes^n_m(V,W) \otimes \otimes^r_s(V,W)$$

and

$$(\alpha_n \otimes \alpha_r) \otimes_1 (\beta_m \otimes \beta_s) = (\alpha_n \otimes \beta_m) \otimes_2 (\alpha_r \otimes \beta_s) .$$

We have subscripted the product symbol $\otimes$ in this equation in order to emphasize the fact that the defining product $\otimes_1$ on the left is between an $(n+r)$-tensor and an $(m+s)$-tensor, while the defined product $\otimes_2$ on the right is between an $(n+m)$-tensor and an $(r+s)$-tensor.

An algebra structure may be placed on $\otimes(V,W)$ by defining a multiplication operation. To this end, let $\alpha^n_m \in \otimes^n_m(V,W)$ and $\beta^r_s \in \otimes^r_s(V,W)$ so that the tensors

$$\alpha = \sum_{n,m} \alpha^n_m \quad , \quad \beta = \sum_{r,s} \beta^r_s$$

are elements of $\otimes(V,W)$. Then the product of two such tensors is given by

$$\alpha\beta = \sum_{\substack{n,m\\r,s}} (\alpha^n_m \otimes \beta^r_s) ,$$

where the symbol $\otimes$ is the same as $\otimes_2$ above. Notice that the multiplication rule implies

$$\begin{aligned}(\alpha_n \otimes \beta_m)(\alpha_r \otimes \beta_s) &= (\alpha_n \otimes \beta_m) \otimes_2 (\alpha_r \otimes \beta_s)\\ &= (\alpha_n \otimes \alpha_r) \otimes_1 (\beta_m \otimes \beta_s)\\ &= (\alpha_n\alpha_r) \otimes (\beta_m\beta_s) .\end{aligned}$$

This relation shows that the algebra $\otimes(V,W)$ is the canonical tensor product of the subalgebras $\otimes V$ and $\otimes W$, or

$$\otimes(V,W) = (\otimes V) \otimes (\otimes W) .$$

The results in this section are adjustments of those which may be found in [14]. Our motivation is, of course, the expansion of functions $f : X \times U \to X$, for X a real vector space of states and U a real vector space of controls.

Formal State Descriptions

Consider a nonlinear state description of the general form

$$\dot{x} = f(x,u)$$

for

$$f : X \times U \to X$$

with X and U real vector spaces, equipped with norm. Let $(\bar{x},\bar{u})$ be a point in $X \times U$, and suppose that

$$D^r f : Z \to L(X \times U,\ldots,X \times U,X)$$

is available for $r = 0,1,2,\ldots,$ with Z open in $X \times U$ and $(\bar{x},\bar{u})$ in Z. Then, formally,

$$f(\bar{x} + x \ , \ \bar{u} + u) = \sum_{k=0}^{\infty} \frac{1}{k!} (D^k f)(\bar{x},\bar{u})(x,u)^{(k)},$$

where $(x,u)^{(k)} = ((x,u),(x,u),\ldots,(x,u))$, the right member having (x,u) k times. It should be recognized that this series could be replaced by a finite number of terms together with a remainder. However, the above representation is adequate for brief illustrative purposes. Space does not permit a discussion of whether, or how, the series acceptably describes the function. Along the same lines, we pass over the related question of how it affects the vector field associated with the differential equation, and therefore its solutions. Instead, we remind the reader that $(D^k f)(\bar{x},\bar{u})$ is a k-linear function on $(X \times U)^k$ to X; and this suggests that we can use tensor algebra to parameterize it. Indeed, denote by $(x,u)^k$ the k-fold tensor product of (x,u) with itself. Then the k-linear function $(D^k f)(\bar{x},\bar{u})$ can be factored uniquely in the manner

$$L_k(\bar{x},\bar{u}) \circ \otimes^k ,$$

where

$$(\otimes^k(X \times U),\ \otimes^k)$$

is a tensor product for k copies of $X \times U$, or what is sometimes called a kth tensorial power for $X \times U$. In this case, the kth parameter map operates in the manner

$$L_k(\bar{x},\bar{u}) : \otimes^k(X \times U) \to X .$$

We have, therefore, that

$$f(\bar{x} + x,\ \bar{u} + u) = \sum_{k=0}^{\infty} \frac{1}{k!} L_k(\bar{x},\bar{u}) \circ \otimes^k (x,u)^{(k)}$$
$$= \sum_{k=0}^{\infty} \frac{1}{k!} L_k(\bar{x},\bar{u})(x,u)^k .$$

Next consider the rearrangement of a term of type

$$L_k(\bar{x},\bar{u})(x,u)^k .$$

Consider, for example, the case $k = 2$, namely

$$(x,u)^2 = (x,u) \otimes (x,u) .$$

Such a form does not relate directly to the structure of the section preceding, which would involve terms of type $x^j \otimes u^m$. However, there is a natural way to convert to that structure. Define projections

$$\pi_u : X \times U \to U \quad ; \quad \pi_x : X \times U \to X ;$$

and injections

$$i_{uu} : U \otimes U \to S \quad ; \quad i_{ux} : U \otimes X \to S ;$$
$$i_{xu} : X \otimes U \to S \quad ; \quad i_{xx} : X \otimes X \to S ;$$

for

$$S = (U \otimes U) \times (U \otimes X) \times (X \otimes U) \times (X \otimes X) .$$

Then we can write

$$\begin{aligned}(x,u) \otimes (x,u) = {} & i_{xx}(\pi_x(x,u) \otimes \pi_x(x,u)) \\ & + i_{xu}(\pi_x(x,u) \otimes \pi_u(x,u)) \\ & + i_{ux}(\pi_u(x,u) \otimes \pi_x(x,u)) \\ & + i_{uu}(\pi_u(x,u) \otimes \pi_u(x,u)) .\end{aligned}$$

If we identify images of the injections with their domains, as for example

$$i_{uu}(U \otimes U) = U \otimes U ,$$

then we can write

$$(x,u) \otimes (x,u) = x \otimes x + x \otimes u + u \otimes x + u \otimes u .$$

According to the conventions of $\otimes(X,U)$, however, discussed in the section preceding, we agree to write

$$u \otimes x = T_{ux,xu}\ x \otimes u$$

for an appropriate isomorphism $T_{ux,xu}$. In that way, we can proceed to

$$\begin{aligned}L_2(\bar{x},\bar{u})(x,u)^2 = {} & L_2(\bar{x},\bar{u})\ x^2 \\ & + L_2(\bar{x},\bar{u})\ x \otimes u \\ & + L_2(\bar{x},\bar{u})T_{ux,xu}\ x \otimes u \\ & + L_2(\bar{x},\bar{u})\ u^2 ,\end{aligned}$$

which we re-notate to (with factorials included)

$$L_{20}(\bar{x},\bar{u})\ x^2 + L_{11}(\bar{x},\bar{u})\ x \otimes u + L_{02}(\bar{x},\bar{u})\ u^2 .$$

In this way, the formal expansion becomes

$$f(\bar{x} + x, \bar{u} + u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} L_{ij}(\bar{x},\bar{u})\ x^i \otimes u^j$$

from which point we can examine the scheduling questions previously raised.

Internal Scheduling

In the section on Multivariable Motivation, we brought attention to four cases of scheduling. With the aid of the ideas foregoing, we would now like to comment on each of these. We shall see that the scheduling idea of our Simple Example Section, while motivating in nature, is not rich enough to embody the complete idea in question. In particular, we have primary interest in the scheduling of parameters as a function of $(\bar{x},\bar{u})$ and not (x,u). Though such issues would have encumbered the Simple Example, we have established now enough background to make the consideration.

Case I

It is sufficient to consider K as a function of r. Notice that r is effectively u for this

ORIGINAL PAGE IS OF POOR QUALITY

case. Accordingly, we have L_{ij} as a mapping

$$L_{ij} : U \to L(\otimes_j^i(X,U),X) \ ,$$

and we may write

$$f(\bar{x} + x, \bar{u} + u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} L_{ij}(\bar{u}) \, x^i \otimes u^j \ .$$

The constraint imposed by Case I is thus a domain restriction on L_{ij}, in the form that the action of L_{ij} depends only upon $\bar{u}$ and not upon $\bar{x}$. If we make the formal expansion

$$L_{ij}(\bar{u}) = \sum_{k=0}^{\infty} L_{ijk} \, \bar{u}^k \ ,$$

in a manner analogous to the steps taken above for $f : X \times U \to X$, then

$$f(\bar{x} + x, \bar{u} + u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} L_{ijk} \, \bar{u}^k \, (x^i \otimes u^j) \ .$$

Assuming distribution, we find that the structure of interest is engendered by

$$L_{ijk} \, \bar{u}^k \, (x^i \otimes u^j) \ .$$

This raises the forms

$$\bar{u}^k \otimes x^i \otimes u^j \ ,$$

which can be referred, by isomorphism, to the previously described tensor algebra by

$$x^i \otimes \bar{u}^k \otimes u^j \ .$$

Case II

We shall assume, for simplicity, that y is equal to x. Again, it is sufficient to consider just one, say K, of the two mappings. In a manner similar to Case I, it can be shown that

$$f(\bar{x} + x, \bar{u} + u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} L_{ijk} \, \bar{x}^k \, (x^i \otimes u^j) \ ,$$

with the corresponding underlying construction

$$\bar{x}^k \otimes x^i \otimes u^j \ .$$

Remark

Although it has not been stated as one of our cases, the generalization of Cases I and II clearly leads to

$$\bar{x}^k \otimes x^k \otimes \bar{u}^k \otimes u^k \ .$$

Case III

This case is quite a bit more complicated than the previous two cases. For simplicity, take L to be the identity and K to be zero. In consideration of G, we note that its input, say u, is a sum of r with a function of the plant output. Suppose, for purposes of G, that H is an identity and that plant output is its state. What we can do is to examine the series combination of P and G, with state

$$z = (x_G, x_P) \in X_G \times X_P \ .$$

If we take u as the input to G, then

$$\dot{z} = f(z,u)$$

with $f : X_G \times X_P \times U \to X_G \times X_P$ and with

$$u = r - \pi_{x_P} z$$

for $\pi_{x_P} : X_G \times X_P \to X_P$ an appropriate projection. Again, we have the basic expansion

$$f(\bar{z} + z, \bar{u} + u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} L_{ij}(\bar{z},\bar{u}) \, z^i \otimes u^j \ ,$$

and we wish to schedule with

$$\bar{r} = \bar{u} + \pi_{x_P} \bar{z} \ .$$

We can reduce this situation to one which is close to that of Case II. Write

$$\begin{aligned} L_{ij}(\bar{z},\bar{u}) &= L_{ij}(\bar{z}, \bar{r} - \pi_{x_P} \bar{z}) \\ &= \tilde{L}_{ij}(\bar{z},\bar{r}) \\ &= \sum_{k=0}^{\infty} \sum_{m=0}^{\infty} \tilde{L}_{ijkm} \bar{z}^k \otimes \bar{r}^m \ . \end{aligned}$$

Then the basic issues involve

$$\bar{r}^m \otimes \bar{z}^k \otimes z^i \otimes u^j \ .$$

The remaining part of this case has to do with H. A similar approach can be applied, if we assume G to be an identity. Here

$$z = (x_P, x_H) \ ,$$

and

$$u = r - \pi_{x_H} z \ ;$$

the results vary essentially only in the meaning of z.

Remark

Assumptions on L, K, G, and H are for convenience only. They can be removed easily by expanding z and the definition of input.

Case IV

Notice that the question of H scheduled on y is the same as H scheduled on its input; and this reduces to Case I. Moreover, the question of G scheduled on y is essentially that of G

ORIGINAL PAGE IS
OF POOR QUALITY

scheduled on a projection of the sum state (x_G, x_H, x_P), which in turn is a question of the composite loop system scheduled on its state, which in turn reduces to a version of Case Ii.

These are a few of the basic concepts which arise in nonlinear scheduling of state equations. We have introduced a viewpoint which seems broad enough to permit discussion. We have not used this framework to solve any new problems, however, but only examined its possibilities for providing frames of reference.

Once internal scheduling has been carried out in an acceptable way, one comes next to the related input/output functions. This is also a large subject, and we give only a sample survey of the issues in the next section.

The Input/Output Connection

In this section we briefly highlight some general results from the literature relating the internal and external constructions for nonlinear dynamical systems. Such results suggest just a few of the approaches to the mathematical bridge necessary for the scheduling discussions of the present paper.

We consider a nonlinear control system

$$\dot{x} = f(x,u)$$

$$y = h(x) \ ,$$

where $f : X \times U \to X$, $h: X \to Y$, for X, U and Y the state, input and output spaces, respectively. With appropriately defined conditions, there is associated with such a system an input-output map from U to Y. Moreover, if we assume a sufficient degree of smoothness for f and h, this input-output map may take on a functional expansion representation commonly known as a Volterra series, first considered by Volterra [15] as early as 1890, and studied later in depth by Weiner.

An important subclass of such systems is that of bilinear systems, which in recent years has itself motivated the study of Volterra series. In [16] the construction of, and realization from, Volterra series representations for bilinear systems is outlined, where it is shown that a necessary and sufficient condition for the existence of bilinear realizations corresponds to that of factorizability of the kernels which characterize the Volterra series expansion. In light of Krener's result [17] for the bilinearization of a rather general class of nonlinear systems, such realization studies take on considerable utility. This latter result is based on a linearization technique used by Brockett [18], introduced first by Carleman [19] in 1932, and used later for similar applications by Bellman and Richardson [20] in 1963. We note here that it is a straightforward exercise to develop this technique analogously in terms of the tensor algebra and corresponding products.

Brockett [21] has expanded these ideas to a more general class of systems for which

$$\dot{x} = g_1(x) + \sum_i u_i g_2(x)$$

$$y = h(x) \ ,$$

where g_1, $g_2 : X \to X$ and $h : X \to Y$ are analytic, and the u_i are components of the vector $u \in U$. Such systems are termed <u>linear analytic</u> and are closed under composition and feedback. It is shown in [21] that the solutions can be expanded in a Volterra series provided that there is no finite escape time. Furthermore, necessary and sufficient conditions for a Volterra series to be realizable as a linear analytic system are given. For a finite Volterra series representation, this will be the case if and only if the kernels are separable.

In [22], Gilbert develops similar results for the general system and for the linear analytic system. The approach is an alternative to the Carleman technique, and uses Frechet power series in the functional expansion. Convergence results of Brockett [21] are utilized for truncated Volterra series.

Yet another approach to these results is introduced in [23] where standard tools from calculus and analysis are employed in studying the existence and uniqueness of Volterra series representations for nonlinear systems. In their existence proof for the Volterra expansion of the class of linear analytic systems, Lesiak and Krener exhibit a procedure for constructing the kernels of the expansion. This construction may be applied to more general systems, where the input enters nonlinearly, by first approximating the system by a bilinear system, using the result in [17]. In the work of Crouch [24], the rich mathematical structures of Lie groups and differential manifolds are explored in relation to the realizations of finite Volterra series for linear analytic systems. A coordinate free development is presented as an extension to the results of [21] and [23].

Truncation

The preceding remarks have made use of series representations for internal models and have discussed relations between internal and input/output models. It is, however, unlikely that one would schedule an entire series in practice, unless that series were simply an alternate form for a function which could be parameterized in finite terms. Thus, although the series format permits a rapid introduction to certain of the underlying structures involved in scheduling, we wish to mention some issues which arise when the series are truncated, or when they had a finite number of terms from the outset.

A principal feature of truncation in an internal or state model is the resulting approximation which occurs in the input/output model. On appropriate normed real vector spaces U and Y of controls and responses, respectively, one might express the input/output function by

$$p : U \to Y$$

and, formally at least, its series expansion by

$$p(\bar{u} + u) = \sum_{k=0}^{\infty} \frac{1}{k!} D^k p(\bar{u}) u^{(k)} \ .$$

The algebra of input/output mappings, under compo-

ORIGINAL PAGE IS
OF POOR QUALITY

sition and pointwise addition, is of ringlike nature, failing only one distributive law. Minor complications arise, moreover, because $U \neq Y$. It is important to realize that similar remarks can be made when the series representation is truncated, as for example in the manner

$$p(\bar{u} + u) \tilde{=} \sum_{k=0}^{m} \frac{1}{k!} D^k p(\bar{u}) u^{(k)} .$$

Call the right member of this equation

$$p_m(\bar{u},u) .$$

Then we have fundamental questions to be addressed insofar as connections are concerned.

Example 1 (Parallel)

Let $p^i : U \to Y,\ i = 1,2$. Then under what conditions does

$$(p^1 + p^2)_m = p_m^1 + p_m^2 \ ?$$

Example 2 (Series)

Let $p^1 : U \to Y$ and $p^2 : Y \to Z$. Then under what conditions does

$$(p^2 \circ p^1)_m = p_m^2 \circ p_m^1 \ ?$$

Of course, the feedback case can also be addressed. Such questions have been addressed in [25,26,27]. Though we have no space remaining here for further detail, an important point is that the questions can be addressed within the framework brought forward in this paper. This offers possibility, then, for determining features of scheduling which involve interconnections in the controller of G, H, L, and K.

Conclusions

Scheduling of local, linear multivariable controllers into global, nonlinear multivariable controllers is a very present tool in a variety of modern control applications. A case in point is gas turbine engine control. See, for example, the companion paper [4].

Despite the entrenchment of scheduling in practice, the theoretical view of such approaches is relatively undeveloped. There are many reasons for this, including the fact that theory is predisposed to give general conditions across many classes of possible application and finds it quite challenging, for the most part, to be general about particular applications. Nonetheless, it seems that much could be gained by a theory/design dialog on the issue of scheduling.

Toward such dialog, we propose in this paper a viewpoint which may be broad enough and deep enough to support such a dialog. The central feature in the viewpoint is the parameterization possibility which is inherent in tensor algebra. For a reasonably general control configuration, we have used the tensor framework to examine four typical scheduling structures. Though much remains to be done in such a dialog, we have found these ideas useful in thinking about the joint theory/application issues which are involved.

Acknowledgment

This work was supported in part by the National Aeronautics and Space Administration under Grant NSG 3048.

References

1. E. Bristol, "Process Control: An Application Theorist's View of Control", IEEE Control Systems Magazine, Vol. 2, No. 1, pp. 3-8, March 1982.
2. E.H. Bristol, "Strategic Design: A Practical Chapter in a Textbook on Control", Proc. 1980 Joint Automatic Control Conf., Vol. I, Paper WA4-A.
3. M.K. Sain, J.L. Peczkowski, and J.L. Melsa, Eds., Alternatives for Linear Multivariable Control. Chicago: National Engineering Consortium, 1978.
4. M.K. Sain and J.L. Peczkowski, "Nonlinear Multivariable Design by Total Synthesis", Proc. 1982 American Control Conference.
5. J.L. Peczkowski, M.K. Sain, and R.J. Leake, "Multivariable Synthesis with Inverses", Proc. 1979 Joint Automatic Control Conf., pp. 375-380.
6. M.K. Sain, B.F. Wyman, R.R. Gejji, P.J. Antsaklis, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part I: Nominal Design", Proc. 1981 Joint Automatic Control Conf., Paper WP-4A.
7. M.K. Sain, P.J. Antsaklis, B.F. Wyman, R.R. Gejji, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part II: Unity Feedback and the Design Morphism", Proc. 1981 IEEE Conf. on Decision and Control, pp. 875-884.
8. J.L. Peczkowski and S.A. Stopher, "Nonlinear Multivariable Synthesis with Transfer Functions", Proc. 1980 Joint Automatic Control Conf., Vol. 1, Pt. WA8-D.
9. J.L. Peczkowski and M.K. Sain, "Scheduled Nonlinear Control Design for a Turbojet Engine", Proc. 1981 IEEE Int. Symp. on Circuits and Systems, pp. 248-251, April 1981.
10. M.K. Sain and J.L. Peczkowski, "An Approach to Robust Nonlinear Control Design", Proc. 1981 Joint Automatic Control Conf., Vol. 2, Pt. FA-3D.
11. J. Dieudonne, Foundations of Modern Analysis. New York: Academic Press, 1960.
12. S. Lang, Analysis. Vols. I and II, Reading: Addison-Wesley, 1969.
13. D.G. Luenberger, Optimization by Vector Space Methods. New York: Wiley, 1969.
14. W.H. Greub, Multilinear Algebra. New York: Springer-Verlag, 1967.
15. V. Volterra, Theory of Functionals and of Integral and Integro-Differential Equations. New York: Dover, 1959.
16. P. d'Alessandro, A. Isidori and A. Ruberti, "Realization and Structure Theory of Bilinear Dynamical Systems", SIAM J. on Control, Vol. 12, No. 3, pp. 517-535, 1974.

17. A.J. Krener, "Linearization and Bilinearization of Control Systems", Proc. 12th Allerton Conf. on Circuit and System Theory, pp. 834-843, October 1974.
18. R.W. Brockett, "On the Algebraic Structure of Bilinear Systems", in Theory and Applications of Variable Structure Systems, R.R. Mohler and A. Ruberti, Eds. New York: Academic Press, pp. 153-168, 1972.
19. T. Carleman, "Application de la theorie des equations integrales singulieres aux equations differentialles de la dynamique", Arkiv for Mathematik, Astronomi och Fysik, Vol. 22B, No. 1, pp. 1-5, 1932.
20. R. Bellman and J.M. Richardson, "On Some Questions Arising in the Approximate Solution of Nonlinear Differential Equations", Quarterly of Applied Mathematics, Vol. 20, No. 4, pp. 333-339, 1963.
21. R.W. Brockett, "Volterra Series and Geometric Control Theory", Automatica, Vol. 12, No. 2, pp. 167-176, 1976.
22. E.G. Gilbert, "Functional Expansions for the Response of Nonlinear Differential Equations", IEEE Trans. on Automatic Control, Vol. AC-22, No. 6, pp. 909-921, 1977.
23. C. Lesiak and A.J. Krener, "The Existence and Uniqueness of Volterra Series for Nonlinear Systems", IEEE Trans. on Automatic Control, Vol. AC-23, No. 6, pp. 1090-1095, 1978.
24. P.E. Crouch, "Dynamical Realizations of Finite Volterra Series", SIAM J. on Control and Optimization, Vol. 19, No. 2, pp. 177-202, 1981.
25. M.K. Sain, "The Quotient Signal Flow Graph for Large Scale Systems", IEEE Trans. on Circuits and Systems, Vol. CAS-25, pp. 781-788, September 1978.
26. M.K. Sain, "Quotient Signal Flowgraphs: New Insights", Proc. 1980 IEEE Int. Conf. on Circuits and Computers", p. 417.
27. M.K. Sain, Introduction to Algebraic System Theory. New York: Academic Press, 1981.

ORIGINAL PAGE IS OF POOR QUALITY

APPENDIX E

"A Computer-Aided Design Package
for Nonlinear Model Applications"

T.A. Klingler
S. Yurkovich
M.K. Sain

Second IFAC Symposium on
Computer Aided Design of Multivariable Technological Systems

Pages 345-353

September 1982

ORIGINAL PAGE IS
OF POOR QUALITY

A COMPUTER-AIDED DESIGN PACKAGE FOR NONLINEAR MODEL APPLICATIONS*

T.A. Klingler, S. Yurkovich, and M.K. Sain

Department of Electrical Engineering, University of Notre Dame,
Notre Dame, Indiana 46556

Abstract. An important aspect of multivariable control system design involves the formulation of reliable mathematical models. Gas turbine engine control systems, with their inherent nonlinearities, provide common practical examples of the need for nonlinear models. In this paper we discuss a computer-aided design package for generation of such nonlinear models, using an approach involving notions of power series and algebraic tensors. Two independent computing systems are employed interactively in the overall process of model formulation, identification, and validation. The package is sufficiently generalized to accommodate any particular nonlinear modeling problem when formulated within the framework of the algebraic tensor scheme.

Keywords. Computer-aided system design; multivariable control systems; modelling; tensor algebra; nonlinear systems; algebraic system theory.

INTRODUCTION

Models have always been an important aspect of applications engineering in the area of multivariable control system design. See for example the work of Kreindler and Rothchild (1976). Practical and industrial examples of the use of models are provided by gas turbine engine control systems, which commonly use models to generate control and response trajectories for various power demands. These models, when scheduled over the operating envelope, can reduce the compensation normally required of the controller, and thus provide the feedback loop with an opportunity to achieve better accuracy in the presence of noise and parametric uncertainties.

The scheduling of feedforward models and feedback compensation typically produces nonlinearities, even if the local models are linear. Accordingly, there is basic interest in fundamental approaches which incorporate nonlinearity at the outset. Such approaches should (1) reduce to the earlier linear schemes for variables with small excursions, (2) be amenable to scheduling, and (3) offer opportunities for determination of parameters from simulation data.

One such approach, investigated by Yurkovich and Sain (1980) and Klingler, Yurkovich, and Sain (1982), uses the notions of power series and algebraic tensors (Sain, 1976) to generate a class of nonlinear models. The important feature of the algebraic tensor is that it provides an organized way of describing the power series expansion formula, lending itself with relative ease to programming on a digital computer. Furthermore, its use allows for the implementation of linear parameter identification techniques.

This paper reports on the development of an interactive computer-aided design package for the formulation, identification, and validation of one particular model structure which uses the above-mentioned tensor approach. The software package, developed using the extensive capabilities of the IBM and DEC Command Procedure Languages with the strengths of FORTRAN and SPEAKEASY, is divided into two segments and tailored to utilize existing computer hardware effectively, as well as to provide the fastest possible user turnaround time. The interactive nonlinear model generation segment is implemented on a Time Sharing Option (TSO) of the IBM 370-168 computer system, where the memory dependent and highly computational routines of the package can benefit from use of the virtual memory and floating point hardware. Once a structured nonlinear model is identified, it is then transfered to the DEC PDP 11/60, where in an interactive environment it can be analyzed and compared to a linear model as well as the true system. In this manner, the user has at his disposal both the graphics capabilities of the video terminal and an electrostatic printer/plotter for the immediate display of data and comparative trajectories.

The remainder of the paper is outlined as follows. First, we briefly discuss notions from analysis and algebra which form the foundation for the tensor approach used in the model formulation. A detailed discussion of the interactive design package is

*This work was supported by the National Aeronautics and Space Administration under Grant NSG 3048.

ORIGINAL PAGE IS
OF POOR QUALITY

then given, followed by a brief discussion of the computational aspects regarding floating point operations in the model simulation phase. We close with an example problem from a turbofan jet engine simulation.

NONLINEAR MODEL FORMULATION

Prior to proceeding to the description of the computer-aided design procedure in the modeling scheme, we outline here some of the prerequisite mathematical issues in a coordinate-free development. Since the treatment is brief, the reader may wish to consult Dieudonne (1960) and Greub (1967) for complete expositions of the topics discussed herein.

Tensor Ideas

We begin with a discussion of abstract derivatives and the calculus of normed vector spaces. Let V and W be normed vector spaces and let Z be open in V. Suppose that $f : Z \to W$ is differentiable at a fixed point p in Z. Then the derivative of $f : Z \to W$ at p is a mapping

$$(Df)(p) : V \to W \tag{1}$$

where

$$Df : Z \to L(V,W) \; ; \tag{2}$$

that is, the derivative mapping in Eq. (1) is an element of the real vector space of *linear* mappings from V to W. Higher order derivatives are defined recursively as

$$(D^r f)(p) = (D(D^{r-1}f))(p) \; , \tag{3}$$

for the positive integer r, provided that the $(r-1)$st derivative is differentiable. Moreover, higher order derivatives are themselves linear mappings according to

$$\begin{aligned} D^2 f(p) &\in L(V,L(V,W)) \; , \\ D^3 f(p) &\in L(V,L(V,L(V,W))) \; , \\ &\vdots \end{aligned} \tag{4}$$

If $L(V^r;W)$ denotes the real vector space of r-linear mappings from V^r to W, it can be shown that there exist isomorphisms

$$\begin{aligned} L(V^2;W) &\to L(V,L(V,W)) \; , \\ L(V^3;W) &\to L(V,L(V,L(V,W))) \; , \\ &\vdots \end{aligned} \tag{5}$$

so that the rth derivative of f at p can be regarded as a mapping from V^r to W. We suppress this isomorphism and consider $D^r f(p)$ as an element of $L(V^r;W)$.

We now use this multilinearity of the derivative mapping to make a connection with notions from algebraic tensors. Let $(\otimes^r V, \otimes^r)$ be a tensor product for r copies of V.

Recall that by the unique factorization property of the tensor product, for every mapping $\psi : V^r \to W$ in $L(V^r;W)$ there exists a mapping $\lambda : \otimes^r V \to W$ in $L(\otimes^r V,W)$ such that $\psi = \lambda \circ \otimes^r$ for $\otimes^r : V^r \to \otimes^r V$ in $L(V^r;\otimes^r V)$. Furthermore, the implication of the unique factorization property is that

$$L(\otimes^r V,W) \to L(V^r;W) \tag{6}$$

is a vector space isomorphism. Thus, via the isomorphisms of Eqs. (5) and (6), the r-linear mapping $D^r f(p) : V^r \to W$ can be composed from a linear mapping $\otimes^r V \to W$ and the universal r-linear tensor product mapping $\otimes^r : V^r \to \otimes^r V$.

State Description

The ideas discussed above are now used to formulate the model structure for a given nonlinear system. We consider systems whose states and inputs are elements of the normed real vector spaces X and U, respectively, and which may be described by the nonlinear ordinary differential equation

$$\dot{x} = f(x,u) \tag{7}$$

for $f : X \times U \to X$. Let $(\bar{x},\bar{u})$ be a fixed point in Z open in $X \times U$, and suppose that $f : X \times U \to X$ is of sufficient smoothness on Z. Formally,

$$f(\bar{x}+x,\bar{u}+u) = \sum_{k=0}^{\infty} \frac{1}{k!} (D^k f)(\bar{x},\bar{u})(x,u)^{(k)} \; , \tag{8}$$

where $(x,u)^{(k)} = ((x,u),(x,u),\ldots,(x,u))$ k times. Due to space limitations we cannot address existence or convergence questions relative to Eq. (8). We note, however, that this series could be replaced by a finite number of terms together with a remainder term in a standard application of Taylor's formula.

According to the preceding discussions we now make the unique factorization

$$D^k f(\bar{x},\bar{u}) = L_k(\bar{x},\bar{u}) \circ \otimes^k \; , \tag{9}$$

where $L_k(\bar{x},\bar{u}) : \otimes^k(X \times U) \to X$ is linear. Denote the k-fold tensor product of (x,u) with itself by $(x,u)^k$ so that, with Eq. (9), we have

$$f(\bar{x}+x,\bar{u}+u) = \sum_{k=0}^{\infty} \frac{1}{k!} L_k(\bar{x},\bar{u})(x,u)^k . \tag{10}$$

Sain and Yurkovich (1982) have shown that the individual terms in the series of Eq. (10) may be rewritten as, for example in the case of $k = 2$,

$$\begin{aligned} \frac{1}{2!} L_2(\bar{x},\bar{u})(x,u)^2 = {} & L_{20}\, x \otimes x + \\ & L_{11}\, x \otimes u + L_{02}\, u \otimes u \; , \end{aligned} \tag{11}$$

where we have suppressed the notation of $(\bar{x},\bar{u})$ on the right side of Eq. (11). In this way Eq. (8) becomes

$$f(\bar{x}+x,\bar{u}+u) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} L_{ij}\, x^{i} \otimes u^{j} \;, \tag{12}$$

forming the structure for the nonlinear model.

Application

In practical applications a truncation approximation of the series in Eq. (12) is considered. In terms of computing, then, the task in the model building scheme is to identify the parameters contained in matrix representations of the L_{ij}. Ordered bases in X, U, $X \otimes X$, $X \otimes U$, and so on, are chosen a priori to be used in constructing a linear least squares identification problem. The ordering algorithm ([illegible]arkovich, 1981) which facilitates this procedure, used in the interactive software package described herein, is easily implemented on a digital computer.

In practice, a differential equation description of the nonlinear system may or may not be available. In either case, the basic strategy involves initial condition and control signal design so that, through data sampling and derivative estimation, a model of the original system of Eq. (7) may be identified. The nonlinear model is required not only to outperform a standard linear model locally about an expansion point, but to establish a larger region of model validity as well.

CATNAP

The intent of this section is to present a detailed discussion of the Computer-Aided Tensor Nonlinear Modeling Applications Package (CATNAP) currently used as a development tool in the formulation, identification, and validation of nonlinear models of the type mentioned above. The structure of CATNAP is based upon ideas from distributed processing and local networking (Tanenbaum, 1981) in which computations are spread over multiple machines. More specifically CATNAP is divided into two segments, each of which is implemented on an independent computing system. These segments are entitled GENERATE and SIMULATE. GENERATE is implemented on a Time Sharing Option (IBM, 1975) of the IBM 370-168 mainframe computer and is used to formulate and identify models, whereas SIMULATE is implemented on the DEC PDP 11/60 computer system and is used to study model validity and performance. Furthermore, both of these segments are highly interactive and contain straightforward input prompts as well as informative error messages.

GENERATE

The GENERATE segment of CATNAP is primarily made up of three routines governed by a higher level supervisor. Figure 1 contains a block diagram depicting the structure of GENERATE.

ORIGINAL PAGE IS
OF POOR QUALITY

GENERATE Supervisory. This supervisory level is written using the command procedure language CLIST (IBM, 1976) and performs the following main functions in sequence:

(1) prompts the user for the name of the desired loader routine to be executed;

(2) passes control to the loader routine defined in (1);

(3) passes control to IDENTIFY; and,

(4) upon user request, passes control to TRANSFER.

In addition to these main functions, certain maintenance roles such as file creation, allocation and deletion are handled by this supervisor.

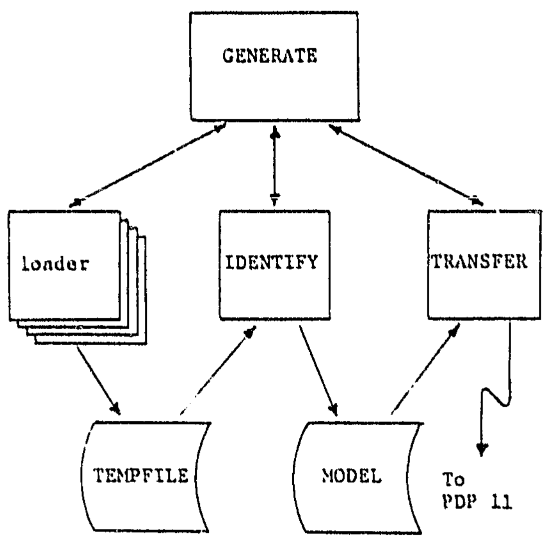

Fig. 1. Block diagram for the CATNAP segment GENERATE.

Loader Routine. Associated with each nonlinear system to be modeled, there exists a loader routine which performs the model formulation task. These routines are stored in a library and are typically written in double precision FORTRAN.

The purpose of any loader routine is to excite the given nonlinear system via initial condition and control input perturbations and to sample the states, inputs and derivative estimates over h selected points in time. The system is then represented by the matrix equation

$$\underset{n\times h}{\dot{X}} = \underset{n\times p}{[L_{10}\ L_{01}\ L_{20}\ L_{11}\ L_{02}\ L_{30}\ \cdots]}\ \underset{p\times h}{X_T} \;. \tag{13}$$

The first n+m rows of the matrix X_T are formed from the sampled values of x and u; the remaining p-(n+m) rows are formed according to the ordering algorithm previously mentioned, which minimizes the number of computations. X is formed by loading derivative estimates for $\dot{x}_1, \dot{x}_2, \ldots, \dot{x}_n$ at the h time points. The number p depends on n and m, and the degree of the truncation approximation. All this data is then stored in TEMPFILE for later use.

Using this approach, CATNAP can accommodate any particular nonlinear modeling problem since the problem specifics are transparent to the remainder of the package. The only requirement is that TEMPFILE contains the appropriate data.

IDENTIFY. After the completion of any chosen loader routine, the program IDENTIFY is executed. IDENTIFY reads the interim data from TEMPFILE and forms a least squares minimization problem which is solved for the desired partitioned matrix containing the desired L_{ij} parameters. These L_{ij} parameters are recorded at the terminal as well as entered into the MODEL data file.

It should be noted here that IDENTIFY is written in the high level language SPEAKEASY, which is based on the concepts of arrays and matrices and processes these as entities. This results in the elimination of the many loops necessary in other programming languages. See the work of Cohen and Pieper (1979). The main reason for employing SPEAKEASY here is that the highly efficient routine SIMEQUAT can be easily used to solve the least squares problem via singular value decomposition, thus reducing the apparent complexity of the problem to a minimum.

TRANSFER. Upon a yes response to a supervisory prompt, the program TRANSFER is submitted batch to the IBM 370-168. TRANSFER is merely a Job Control Language (JCL) deck which sends a copy of the file MODEL, containing the L_{ij} parameters, to the DEC PDP 11/60 computing system by the way of a Remote Job Entry port, and stores it in the nonlinear model library. An excellent account of JCL can be found in Brown (1977).

SIMULATE

Shifting our concern away from the discussion of GENERATE, we now focus our attention on the SIMULATE segment of CATNAP. Basically, two routines plus a supervisor comprise the structure of SIMULATE. Figure 2 offers an illustration of this structure to supplement the following presentation.

SIMULATE Supervisory. Written in the form of an Indirect Command File (DEC, 1979), this supervisor allows the user to:

(1) create new simulator routines;

(2) execute existing simulator routines; and,

(3) execute VERSATEC which produces hardcopy plots.

As earlier, this supervisor performs a number of file maintenance duties in addition to the above functions.

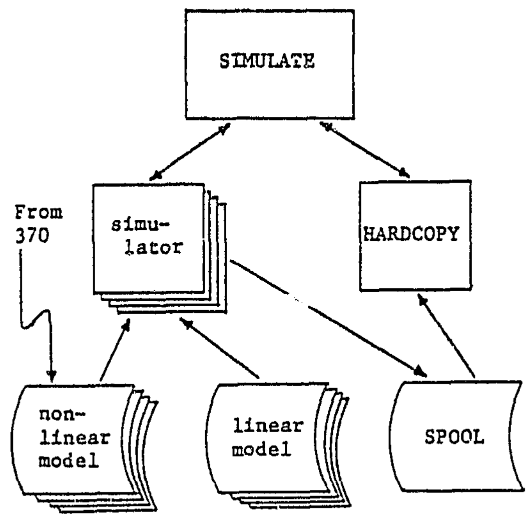

Fig. 2. Block diagram for the CATNAP segment SIMULATE.

Simulator Routine. A FORTRAN simulator routine usually exists for each nonlinear modeling problem studied; however, only one subroutine in that program is altered among versions, and that is the application subroutine TRUES. The remainder of the program stays unchanged. TRUES contains the true system representation of the nonlinear system being modeled, and is used extensively in comparison studies. Because of the number of TRUES subroutines that exist, a library has been created to store the various simulator routines.

The execution of a particular simulator routine can be divided into three steps: (1) problem configuration; (2) systems integration; and (3) solution display.

The first of these steps requires the user to decide which of the available systems, true solution, linear model and/or nonlinear model, should be included in the session configuration. When a model is chosen, the user is asked to enter the name of the desired model. That model is then read into the simulator from the appropriate library. The linear models used in CATNAP are generally identified by standard techniques and are available for use in comparison studies.

Next, the user is prompted for various integration parameters such as stepsize and upper time limit as well as initial condi-

ORIGINAL PAGE IS
OF POOR QUALITY

tions, input amplitudes and frequencies. The configured systems are then integrated and the solutions are sampled at 100 points, evenly spaced in time.

Finally, to assist in the data analysis, a number of options are available to the user. They include:

(1) printing the solutions on the Versatec;

(2) displaying the comparative trajectories on the Tektronix graphics terminal;

(3) writing the trajectory solutions to SPOOL for hardcopy plotting at a later time; and,

(4) solving the configuration for another set of initial conditions and control inputs.

The use of these options provides a powerful yet flexible capability for the study of model performance and validity. Furthermore, when all three systems are included in the configuration, an additional error criterion is generated and used in testing the performance of the nonlinear model versus that of the linear model.

Let ε_i^N denote the absolute maximum error in the nonlinear model solution, as compared to the true simulation solution, over the time range of simulation for the ith state variable. Similarly, we define ε_i^L for the linear model error. Then ε_i is the comparison $\varepsilon_i^N - \varepsilon_i^L$. Thus, if ε_i is negative, the nonlinear model has exhibited a smaller maximum absolute error in the ith state, and in that sense has outperformed the linear model.

VERSATEC. The routine VERSATEC, written in FORTRAN, reads the trajectory solutions from SPOOL and records at the Versatec printer/plotter, a data sheet corresponding to each plot set which follows. The comparative trajectories themselves are then plotted.

MODEL SIMULATION

In this section we comment on the efficiency of the model structure discussed above by studying the number of floating point operations (FLOPs) necessary in typical simulations. It is common practice in computer architecture to design processors which require no extra time for floating point additions calculated simultaneously with multiplications. Thus, we concern ourselves primarily with the latter, and by FLOPs we will mean multiplies. Since the largest burden of the computer in the simulation process is the actual numerical integration of model differential equations, we will analyze only that portion of the simulation.

The system to be considered takes the form of Eq. (13), or

$$\dot{x} = Lz ,$$

where x is the n-vector of states, L the parameter matrix, and z the p-vector of ordered monomial terms derived from the various symmetric products of x and u, the m-vector of inputs (Yurkovich and Sain, 1980). The least number of multiplications required to construct z is merely $p-n+m$, or the total number of terms which involve products. This number is given by

$$p-(n+m) = \sum_{i=2}^{d} \left\{ \binom{n+i-1}{i} + \binom{m+i-1}{i} + \left[\sum_{j=1}^{i-1} \binom{n+(i-j)-1}{(i-j)} \binom{m+j-1}{j} \right] \right\} ,$$

where d is the model degree.

Assuming the use of a fourth-order integration routine, the number of FLOPs necessary to formulate and integrate the system as embodied by the model, across one integration time step, is $4(n)(p)$. As an illustration consider a four-state, three-input model.[1] Suppose, for simplicity, that 100 integration time steps is the equivalent of one second in real time. This translates roughly to 0.25 million FLOPs per second for a degree-3 model (an approximation which retains terms up to and including the third degree). While there are many other obvious considerations involved in real time simulation, this number is well within the bounds dictated by state-of-the-art computation speeds of ten million FLOPs per second.

EXAMPLE

In the example to follow attention will center around NASA's QCSEE ("Quixie")---Quiet, Clean, Shorthaul Experimental Engine. Wise (1974) provides an excellent overview of the QCSE engine program. QCSEE is designed specifically for powered-lift, short-haul aircraft, and incorporates several new concepts not all currently used on turbofans to achieve operational efficiency in a quiet, clean manner.

QCSEE APPLICATION

For this nonlinear modeling problem, a complex eight-state, three-control digital simulation of the QCSE engine is employed (Mihaloew, 1981). Using this digital deck as the system representation, a loader routine, QCSELOAD, is constructed to formulate a reduced order four-state, three-control analytical model. The engine states chosen are the combustor discharge pressure (P4GS), the core nozzle pressure (P8GS), the fan speed (NL), and the compressor speed (NH). The control inputs used are the fuel flow

[1] This represents a typical model as investigated by Klingler, Yurkovich, and Sain (1982).

ORIGINAL PAGE IS
OF POOR QUALITY

(WFM), the exhaust nozzle area (A18) and the fan pitch angle (BETAF). In a similar way, the simulator routine QIXSIM is built using QCSEE as the true system in the subroutine TRUES.

For the results presented herein, two models have been formulated using QCSELOAD at 92% power demand: a second-degree nonlinear model, and a first-degree linear model. Both formulations are made using 1% steady state perturbations in the state and control variables. Furthermore, the control inputs are manipulated so that cosinusoidal signals are inserted. The observed states and inputs are sampled over a six second interval, and the difference between these values and the corresponding equilibrium values, together with the ordered tensor product terms and state derivative values comprise the data necessary for the identification. The model parameters are easily computed from IDENTIFY and then sent to the PDP 11/60 via TRANSFER.

Using the capabilities of QIXSIM and VERSATEC, several validation studies have been completed to date, all yielding satisfactory results. Figures 3-7 contain a representative plot set from VERSATEC illustrating the model performance for a particular input set, as well as the graphical capabilities of CATNAP. Table 1 contains a variable ledger for Figures 3-7.

CONCLUSION

The importance of nonlinear modeling in multivariable control systems could hardly be overemphasized. And the applications side of the problem has benefited greatly with the advent of expanded and more versatile computing environments.

Rarely does it happen, though, that one computing system can accommodate all requirements placed on it, particularly when plagued by multiple users demanding unlimited access. It often happens, however, that the capabilities of several computing systems are at ones disposal, each with various features to offer. In this case schemes employing the notions of distributed processing and local networking become extremely useful.

We have presented one such scheme in the form of an interactive computer-aided design package for a specific nonlinear modeling problem. The software package facilitates the analysis of complex problems, with relative ease to the user, from the initial model formulation and identification stage through to the model testing and validation studies. Series ideas and algebraic tensors are the main vehicles in the model formulation. The importance of the tensor approach lies in its parametric possibilities, and ongoing research is currently underway to exploit further the richness of such structures.

REFERENCES

Brown, G.D. (1977). System 370 Job Control Language. John Wiley and Sons, New York.

Cohen, S., and S. Pieper (1979). The Speakeasy III Reference Manual. Speakeasy Computing Corporation, Chicago.

DEC (1979). RSX-11M/M-Plus MCR Operations Manual, AA-H263A-TC. Chapter 5, Digital Equipment Corp., Maynard, MA.

Dieudonne, J. (1960). Foundations of Modern Analysis. Academic Press, New York.

Greub, W.H. (1967). Multilinear Algebra. Springer-Verlag, New York.

IBM (1975). OS/VS2 TSO Terminal User's Guide, GC28-0645-2, IBM Corporation, Poughkeepsie, N.Y.

IBM (1976). OS/VS2 TSO Command Language Reference, GC28-0646-3. IBM Corporation, Poughkeepsie, N.Y.

Klingler, T.A., S. Yurkovich, and M.K. Sain (1982). An Application of Tensor Ideas to Nonlinear Modeling of a Turbofan Jet Engine. Proc. Thirteenth Pittsburgh Conf. on Modeling and Simulation, to appear.

Kreindler, E., and D. Rothchild (1976). Model Following in Linear Quadratic Optimization. AIAA J., 14, pp. 835-842.

Mihaloew, J.R. (1981). A Nonlinear Propulsion System Simulation Technique for Piloted Simulators. Proc. Twelfth Pittsburgh Conf. on Modeling and Simulation, pp. 1423-1427.

Sain, M.K. (1976). The Growing Algebraic Presence in Systems Engineering: An Introduction. Proc. IEEE, 64, pp. 96-111.

Sain, M.K., and S. Yurkovich (1982). Controller Scheduling: A Possible Algebraic Viewpoint. Proc. 1982 American Control Conf., to appear.

Tanenbaum, A.S. (1981). Computer Networks. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Wise, C.E. (1974). Turbofan of the Future. Machine Design, pp. 20-25.

Yurkovich, S. (1981). Application of Tensor Ideas to Nonlinear Modeling and Control. M.S. Thesis, Elec. Engrg. Dept., Univ. of Notre Dame, Notre Dame, IN.

Yurkovich, S., and M.K. Sain (1980). A Tensor Approach to Modeling of Nonhomogeneous Nonlinear Systems. Proc. Eighteenth Allerton Conf. on Communication, Control, and Computing, pp. 604-613.

TABLE 1 Variable Ledger for Figures 3-7

o : True Engine Response Δ : Linear Model Response * : Nonlinear Model Response	
x_1: P4GS (psi) x_2: P8GS (psi) x_3: NL (rpm) x_4: NH (rpm)	u_1: WFM (lb_m/hr) u_2: A18 (in^2) u_3: BETAF (deg.)

ORIGINAL PAGE IS
OF POOR QUALITY

```
*************** PROBLEM SUMMARY ***************
    CONFIGURATION:  TRUE, LINEAR & NONLINEAR
    NUMBER OF STATES:  4
    NUMBER OF CONTROLS:  3
    LENGTH OF TENSOR TERM VECTOR:  35
    DEGREE OF APPROXIMATION:  2
***********************************************
```

SOLUTION PARAMETERS:
****

INTEGRATION STEPSIZE: Ø.Ø1Ø
UPPER TIME LIMIT OF INTEGRATION: 2.ØØØ
NUMBER OF PLOT POINTS: 1ØØ
SPACING BETWEEN PLOT POINTS: Ø.Ø2Ø

STATE NUMBER	INITIAL CONDITION	ERROR CRITERION
1	-Ø.Ø75	-Ø.533E+ØØ
2	Ø.ØØ1	-Ø.915E-Ø2
3	-Ø.25Ø	-Ø.698E+Ø1
4	Ø.5ØØ	-Ø.111E+Ø2

CONTROL NUMBER	AMPLITUDE	FREQUENCY (CYCLES/SEC)
1	74.29Ø	2.1ØØ
2	-21.2ØØ	1.ØØØ
3	-Ø.239	1.5ØØ

Fig. 3. Sample data sheet for the QCSEE example

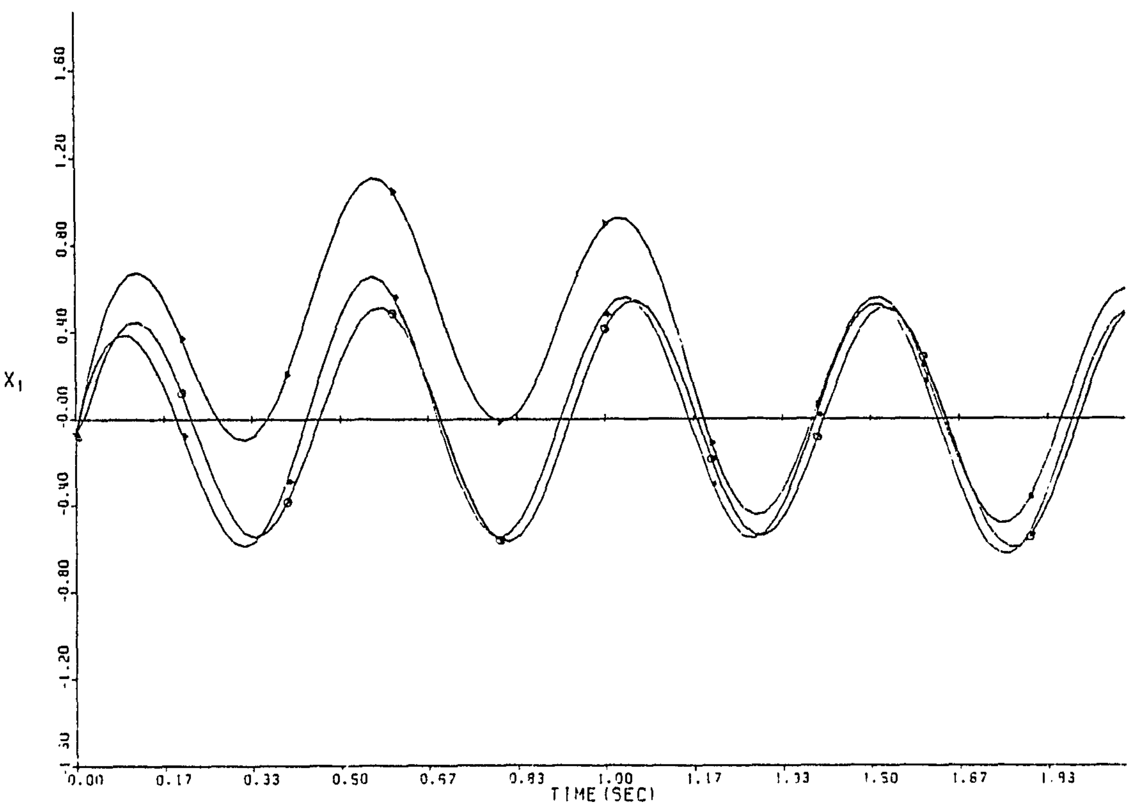

Fig. 4. Comparative solutions: Fig. 3, state 1.

ORIGINAL PAGE IS
OF POOR QUALITY

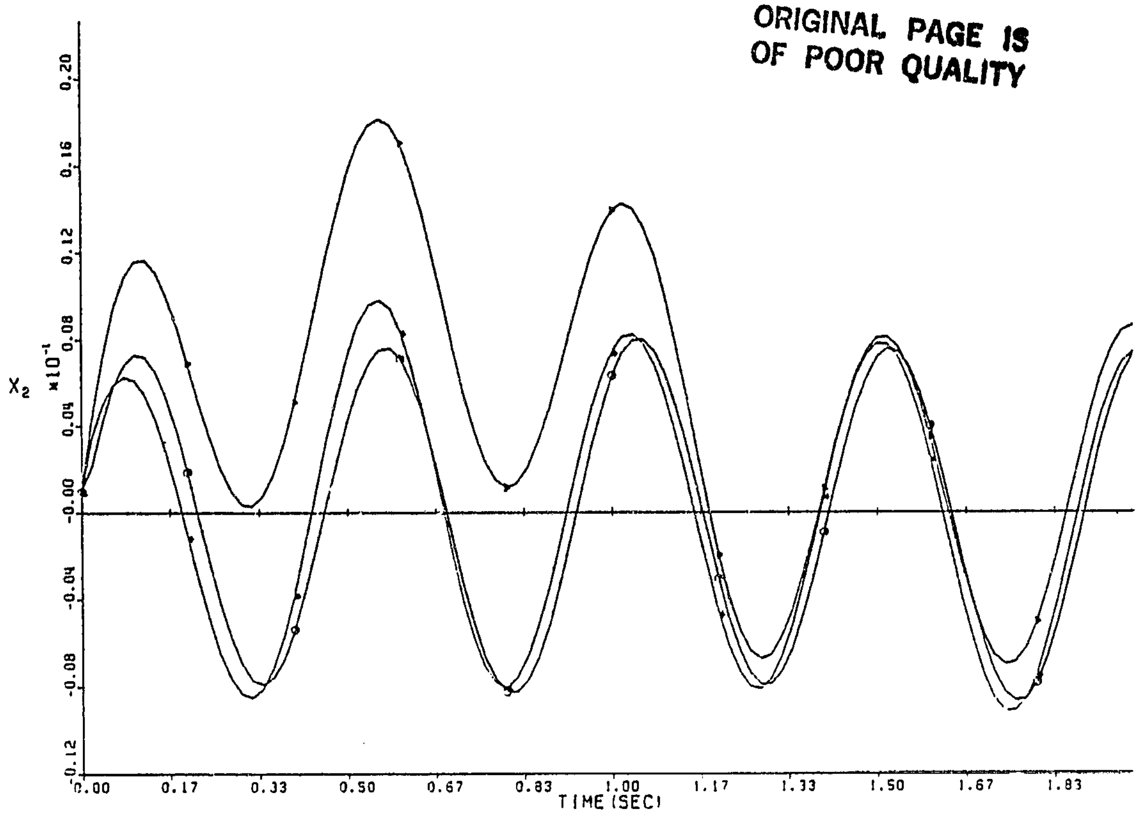

Fig. 5. Comparative solutions: Fig. 3, state 2.

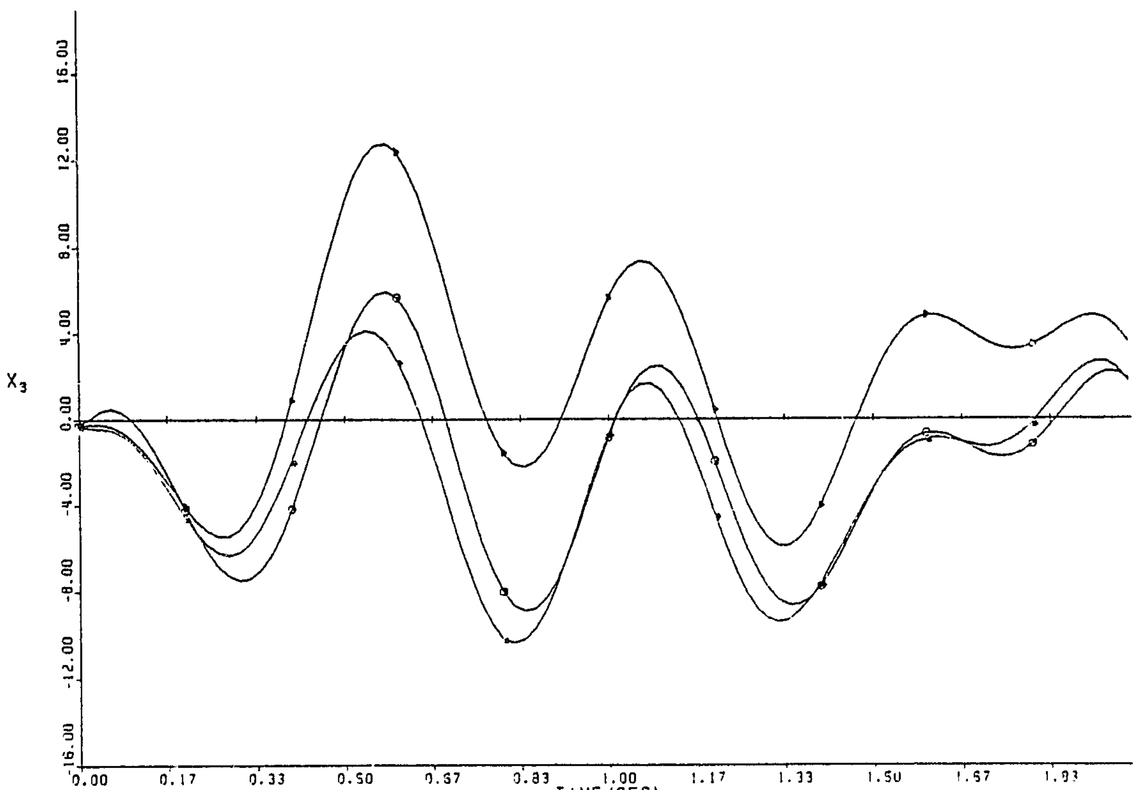

Fig. 6. Comparative solutions: Fig. 3, state 3.

ORIGINAL PAGE IS
OF POOR QUALITY

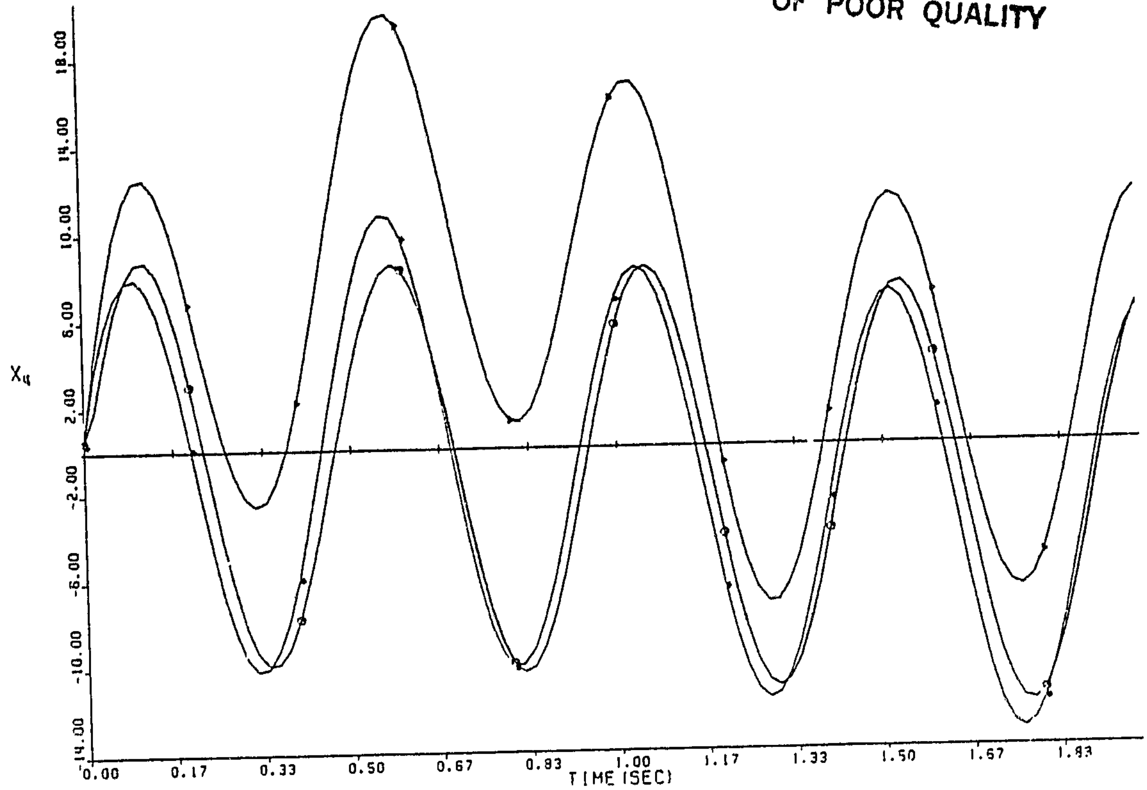

Fig. 7. Comparative solutions: Fig. 3, state 4.

APPENDIX F

Reference List on Total Synthesis Problem

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

Reference List on Total Synthesis Problem

1. J.L. Peczkowski and M.K. Sain, "Linear Multivariable Synthesis with Transfer Functions", in Alternatives for Linear Multivariable Control, M.K. Sain, J.L. Peczkowski, and J.L. Melsa, Eds. Chicago: National Engineering Consortium, 1978, pp. 71-87.

2. J.L. Peczkowski, M.K. Sain, and R.J. Leake, "Multivariable Synthesis with Inverses", Proceedings Joint Automatic Control Conference, pp. 375-380, 1979.

3. J.L. Peczkowski, "Multivariable Synthesis with Transfer Functions", Proceedings Propulsion Controls Symposium, pp. 111-128, May 1979.

4. R.J. Leake, J.L. Peczkowski, and M.K. Sain, "Step Trackable Linear Multivariable Plants", International Journal of Control, Volume 30, Number 6, pp. 1013-1022, December 1979.

5. M.K. Sain, A. Ma, and D. Perkins, "Sensitivity Issues in Decoupled Control System Design", Proceedings Southeastern Symposium on System Theory, pp. 25-29, May 1980.

6. J.L. Peczkowski and M.K. Sain, "Control Design with Transfer Functions, an Application Illustration", Proceedings Midwest Symposium on Circuits and Systems, pp. 47-52, August 1980.

7. M.K. Sain and A. Ma, "Multivariable Synthesis with Reduced Comparison Sensitivity", Proceedings Joint Automatic Control Conference, Paper WP-8B, August 1980.

8. J.L. Peczkowski and S.A. Stopher, "Nonlinear Multivariable Synthesis with Transfer Functions", Proceedings Joint Automatic Control Conference, Paper WA-8D, August 1980.

9. M.K. Sain, R.M. Schafer, and K.P. Dudek, "An Application of Total Synthesis to Robust Coupled Design", Proceedings Allerton Conference on Communication, Control, and Computing, pp. 386-395, October 1980.

10. J.L. Peczkowski and M.K. Sain, "Scheduled Nonlinear Control Design for a Turbojet Engine", Proceedings IEEE International Symposium on Circuits and Systems, pp. 248-251, April 1981.

11. M.K. Sain and J.L. Peczkowski, "An Approach to Robust Nonlinear Control Design", Proceedings Joint Automatic Control Conference, Paper FA-3D, June 1981.

12. R.R. Gejji, "On the Total Synthesis Problem of Linear Multivariable Control", Ph.D. Thesis, Department of Electrical Engineering, University of Notre Dame, May 1980.

ORIGINAL PAGE IS OF POOR QUALITY

13. M.K. Sain, B.F. Wyman, R.R. Gejji, P.J. Antsaklis, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part I: Nominal Design", Proceedings Joint Automatic Control Conference, Paper WP-4A, June 1981.

14. B.F. Wyman and M.K. Sain, "The Zero Module and Essential Inverse System", IEEE Transactions on Circuits and Systems, Volume CAS-27, Number 2, pp. 112-126, February 1981.

15. P.J. Antsaklis and M.K. Sain, "Unity Feedback Compensation of Unstable Plants", Proceedings IEEE Conference on Decision and Control, pp. 305-308, 1981.

16. M.K. Sain, P.J. Antsaklis, B.F. Wyman, R.R. Gejji, and J.L. Peczkowski, "The Total Synthesis Problem of Linear Multivariable Control, Part II: Unity Feedback and the Design Morphism", Proceedings 20th IEEE Conference on Decision and Control, pp. 875-884, 1981.

17. J.L. Peczkowski and M.K. Sain, "Nonlinear Multivariable Design by Total Synthesis", Proceedings American Control Conference, 1982.

18. M.K. Sain and R.M. Schafer, "A Computer-Assisted Approach to Total Feedback Synthesis", Proceedings American Control Conference, 1982.

APPENDIX G

SOFTWARE DESCRIPTION FOR SECTION IV

In this appendix, we provide the necessary documentation of the software that was used for both the first and second order controller problems. The computer programs are written in the high-level language SPEAKEZY [23] for use on the IBM 370/168 at the University of Notre Dame. The first program is FIRORDA, which provides the first order analysis of the example problem. This procedure, as was demonstrated, is identical to the solution of the Riccati equation in matrix form and substitution of the optimal control into the system equations. The program first integrates the Riccati equation using the modified Euler method described in [28]. This provides the solution for the optimal value term $V_2(t)$ and subsequently the first term in the controller expansion $K_1(t)$. A fourth-order Runge-Kutta integration routine then solves for the state variable $x(t)$ after appropriate substitutions have been made for the controller term. The final part of the program provides for the reduction of the array size (if necessary) and the plotting of both the regulated state variables and also the control variables.

The second order analysis of the example problem required two programs. The first program, called COMATRCS, provided for the calculation of the coefficient matrices that appeared in equations (4.4) and (4.5). In particular, the program calculated those coefficient matrices that remain constant over the integration interval. These matrices may be listed as follows:

ORIGINAL PAGE IS
OF POOR QUALITY

$$CM1 = (A_{11} \otimes 1_x)S^1_{21} + (1_x \otimes A_{11}) ,$$

$$CM2 = (A_{02} \otimes 1_x)S^1_{12} + (1_x \otimes A_{02}) ,$$

$$CM3 = (A_{01} \otimes 1_x^{\,2})S^2_{21} + (1_x \otimes A_{01} \otimes 1_x)S^1_{21} + (1_x^{\,2} \otimes A_{01}) ,$$

$$CM4 = A_{10} \oplus A_{10} \oplus A_{10} ,$$

$$CM5 = (A_{01} \otimes 1_x)S^1_{11} + (1_x \otimes A_{01}) ,$$

$$CM6 = (A_{20} \otimes 1_x) + (1_x \otimes A_{20}) .$$

The program that actually does the second order analysis is called SECORDA. This routine integrates to get $V_3(t)$, again using the modified Euler method, and subsequently calculates $K_2(t)$. Using the results from the FIRORDA program, the second order controller may be generated. The system is then integrated to yield the state variables by using a fourth order Runge-Kutta integration routine. The results are then plotted in the final section of the program. In the solution of equation (4.4) for $K_2(t)$ in terms of $V_3(t)$, we define

$$VM1 = 1_x \otimes \pi_s \,(K_1 \otimes 1_u) ,$$

which needs to be updated as $K_1(t)$ varies. Also needed are the Kronecker product matrices

$$E1 = 1_x^{\,2} \otimes K_1 ,$$

$$E2 = 1_x \otimes K_2 ,$$

$$E3 = 1_x \otimes K_1 \otimes K_1 ,$$

$$E4 = K_1 \otimes K_2 ,$$

$$E5 = K_2 \otimes K_1 .$$

Like the VM1 term, these matrices also must be updated at each integration step.

ORIGINAL PAGE IS
OF POOR QUALITY

The final section of this appendix lists the time-varying control gains and the optimal value expression terms. In each case, the expressions are listed as functions of time, which appears in the first column. The array KA lists $K_1(t)$; the array K2A lists $K_2(t)$; the array VA lists $V_2(t)$; and the array V3A lists $V_3(t)$. In all cases, we have assumed an integration interval of 0 to 5 seconds and a stepsize of 0.05 seconds.

ORIGINAL PAGE IS
OF POOR QUALITY

EDITING FIRORDA

```
PROGRAM
$
$ THIS PROGRAM PROVIDES THE FIRST ORDER ANALYSIS OF THE EXAMPLE
$PROBLEM, THAT IS, THE CALCULATION OF V2(T) AND K1(T).  THE
$INTERVAL IS ASSUMED TO BE FROM T=0 TO T=T1, WITH INCREMENTS DT.
$INTEGRATION PROCEEDS BACKWARDS FROM AN INITIAL VALUE OF V(T1)=
$M2 UNTIL A FINAL VALUE OF V(0), WHICH IS FREE.  THIS RESULT IS
$IDENTICAL TO THE RICCATI EQUATION FOR THE LINEAR-QUADRATIC CASE,
$EXCEPT THAT WE ASSUME THE EXISTENCE OF THE C MATRIX, WHICH WEIGHTS
$THE (1,1) TERM OF THE L(X,U,.)EXPANSION.  ONCE THE VALUES OF THE
$TIME VARYING K1 MATRIX HAVE BEEN CALCULATED, THEY ARE STORED IN
$AN ARRAY KA, WHICH ALSO CONTAINS THE TIME VARIABLE.  THESE GAINS
$WILL BE USED IN THE NEXT PART OF THE PROGRAM TO CALCULATE THE
$TERMS OF THE FIRST ORDER CONTROLLER.
$
$RETRIEVE DATA FILES, INITIALIZE MATRICES AND LOAD
$
GET A01MAT; GET A10MAT; GET M2VEC
GET RMAT; GET QMAT; GET CMAT
A01=MAT(2,2'); A10=MAT(2,2'); M2=MAT(2,2')
R=MAT(2,2'); C=MAT(2,2'); Q=MAT(2,2')
LOADDATA(A01,A01MAT); LOADDATA(A10,A10MAT)
LOADDATA(R,RMAT); LOADDATA(C,CMAT); LOADDATA(Q,QMAT)
LOADDATA(M2,M2VEC)
REQUEST T1,DT
S=(T1/(-DT))+1
X=ARRAY(2'); U=ARRAY(2'); KA=ARRAY(S,5'); VA=ARRAY(S,5')
V=M2
AM=A10-.5*A01*(1/R)*TRANSP(C)
R1=S
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FOR T=T1,0,DT
$
$CALCULATE K=K1 AND UPDATE FUNCTIONAL ARRAYS
$
K=-(1/R)*(TRANSP(A01)*V+.5*TRANSP(C))
VA(R1,1)=T, VA(R1,2)=V(1,1), VA(R1,3)=V(1,2), VA(R1,4)=V(2,1)
VA(R1,5)=V(2,2), KA(R1,1)=T, KA(R1,2)=K(1,1), KA(R1,3)=K(1,2)
KA(R1,4)=K(2,1), KA(R1,5)=K(2,2)
$
$CALCULATE V=V2 FROM MATRIX RICCATI EQUATION
$
VMP=V+(DT/2)*(-V*AM-TRANSP(AM)*V+V*A01*(1/R)*TRANSP(A01)*V-
&Q+.25*C*(1/R)*TRANSP(C))
V=V+DT*(-VMP*AM-TRANSP(AM)*VMP+VMP*A01*(1/R)*TRANSP(A01)*VMP-
&Q+.25*C*(1/R)*TRANSP(C))
$
$UPDATE THE R1 INDEX AND T TO CONTINUE INTEGRATION.  STORE KA, VA
$
R1=R1-1
ENDLOOP T
KEEP KA, KEEP VA
PRINT"ENTER CHOICE=1 TO CONTINUE OR CHOICE=0 TO STOP AND
&DISPLAY KA AND VA"
REQUEST CHOICE
IF(CHOICE.EQ.0) GO TO L6
$
$ THE NEXT SECTION PERFORMS THE NECESSARY INTEGRATIONS
$AFTER THE SUBSTITUTION OF THE OPTIMAL CONTROL U(T)IS MADE INTO THE
$SYSTEM EQUATIONS.  THE METHOD OF NUMERICAL INTEGRATION USED IS A
$FOURTH ORDER RUNGE-KUTTA ROUTINE USING KUTTA'S COEFFICIENTS.  THIS
```

```
$PART OF THE PROGRAM PRODUCES THE ARRAYS X1A AND X2A, WHICH CONTAIN
$THE STATE VARIABLES AND THE ARRAYS U1A AND U2A, WHICH CONTAIN
$THE CONTROLLER COMPONENTS, AGAIN SEQUENCED BY TIME.  THESE ARRAYS
$WILL BE USED LATER TO GENERATE THE NECESSARY TIME PLOTS.
$
PRINT "OPTIMAL VALUE INTEGRATION COMPLETE"
$
$INITIALIZE MAIN COUNT (MCT), RUNGE KUTTA COEFFICIENT ARRAYS
$RKC1 AND RKC2, FUNCTION ARRAYS X1A, X2A, U1A, U2A, AND TIME
$
RUNGE:  MCT=1; PERINDX=0
REQUEST X10,X20
H=-DT;RKC1=ARRAY(5:);RKC2=ARRAY(5:)
X1A=ARRAY(S:); X2A=X1A; U1A=X1A; U2A=X1A
TA=X1A
X(1)=X10;X(2)=X20;T=0
L5:  XP=X; COUNT=1
$
$UPDATE CONTROL FUNCTION AND SUBSTITUTE INTO SYSTEM EQUATIONS
$
L2:  U(1)=KA(MCT,2)*XP(1)+KA(MCT,3)*XP(2)
U(2)=KA(MCT,4)*XP(1)+KA(MCT,5)*XP(2)
F1A=U(2)*(EXP(XP(1)*XP(2))+EXP(-XP(1)*XP(2)))*.5-EXP(2*U(1))
&*(EXP(2*XP(1))-EXP(-2*XP(1)))*.5
F1B=-3*(EXP(XP(2))-EXP(-XP(2)))*.5
F1=F1A+F1B
F2A=EXP(U(1)*U(2))*(EXP(XP(1))-EXP(-XP(1)))*.5-EXP(U(1))*U(1)*
&(EXP(XP(1)*XP(1))+EXP(-XP(1)*XP(1)))*.5
F2B=(EXP(XP(2))-EXP(-XP(2)))*.5
F2=F2A+F2B
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
$
$CALCULATE RUNGE KUTTA COEFFICIENTS
$
RKC1(COUNT)=F1*H; RKC2(COUNT)=F2*H
IF(COUNT.EQ.4) GO TO L4
$
$UPDATE PROVISIONAL X FOR NEXT COEFFICIENT
$
XP(1)=X(1)+.5*RKC1(COUNT)
XP(2)=X(2)+.5*RKC2(COUNT)
COUNT=COUNT+1
IF(COUNT.EQ.4) GO TO L3
GO TO L2
L3:  XP(1)=X(1)+RKC1(3)
XP(2)=X(2)+RKC2(3)
GO TO L2
$
$CALCULATE DELTA X
$
L4:  RKC1(5)=(RKC1(1)+2*RKC1(2)+2*RKC1(3)+RKC1(4))/6
RKC2(5)=(RKC2(1)+2*RKC2(2)+2*RKC2(3)+RKC2(4))/6
$
$UPDATE FUNCTION ARRAYS U1A, U2A, X1A, AND X2A
$
X1A(MCT)=X(1); X2A(MCT)=X(2)
U1A(MCT)=KA(MCT,2)*X(1)+KA(MCT,3)*X(2)
U2A(MCT)=KA(MCT,4)*X(1)+KA(MCT,5)*X(2)
TA(MCT)=T
$
$UPDATE PERFORMANCE INDEX
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
$
UV=VECTOR(2:U1A(MCT),U2A(MCT)); XV=VFAM(X)
PERINDX=PERINDX+(XV*Q*XV+UV*R*UV+XV*C*UV)
$
$UPDATE STATE VARIABLES FOR NEXT GO AROUND
$
X(1)=X(1)+RKC1(5)
X(2)=X(2)+RKC2(5)
$
$UPDATE MCT, TIME FOR NEXT GO AROUND
$
MCT=MCT+1; T=T+H
IF(MCT.NE.S+1) GO TO L5
PERINDX=.5*(XV*M2*XV+PERINDX)
PRINT PERINDX
PRINT "STATE INTEGRATION COMPLETE"
$
$ WE ARE NOW READY TO BEGIN THE MECHANICS OF PLOTTING THE STATE
$AND CONTROL VARIABLES VERSUS TIME.  BECAUSE OF THE LARGE SIZES OF
$THE ARRAYS, IT WILL BE USEFUL IF WE SELECT ONLY A CERTAIN NUMBER
$OF POINTS FROM EACH ARRAY THAT ACTUALLY WILL BE PLOTTED.  THIS WILL
$BE DONE BY SPECIFYING THE PARAMETER SEL.  ONCE THESE SMALLER ARRAYS
$HAVE BEEN FORMED, THE FINAL STEP IN THE PROCEDURE IS THE PLOTTING
$OF FOUR GRAPHS - X(1), X(2), U(1), AND U(2) ALL VERSUS TIME
$OVER THE INTERVAL, WHICH IS USER SPECIFIED FROM T=0 TO T=T1.
$
GRONLY:  REQUEST SEL
FV=1+(SEL)*INTPART(S/SEL)
TAPL=ARRAY(1:)
X1APL=TAPL; X2APL=TAPL; U1APL=TAPL; U2APL=TAPL
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
INDX=1
$
$FORM REDUCED SIZE ARRAYS FOR PLOTTING
$
IF(FRAC(S/SEL).EQ.0) FV=(S+1)-SEL
FOR J=1,FV,SEL
TAPL(INDX)=TA(J)
X1APL(INDX)=X1A(J)
X2APL(INDX)=X2A(J)
U1APL(INDX)=U1A(J)
U2APL(INDX)=U2A(J)
INDX=INDX+1
ENDLOOP J
$
$PLOTTING BEGINS HERE
$
GRAPHICS(TEK4010)
SETXLABEL 'TIME IN SECONDS'
SETYLABEL 'X1'
SETTITLE 'STATE VARIABLE X1 VS. TIME'
LINECODE=1
GRAPH(X1APL,TAPL)
HARDCOPY
SETYLABEL 'X2'
SETTITLE 'STATE VARIABLE X2 VS. TIME'
GRAPH(X2APL,TAPL)
HARDCOPY
SETYLABEL 'U1'
SETTITLE 'CONTROL VARIABLE U1 VS. TIME'
GRAPH(U1APL,TAPL)
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR Q...

```
HARDCOPY
SETYLABEL 'U2'
SETTITLE'CONTROL VARIABLE U2 VS. TIME'
GRAPH(U2APL,TAPL)
HARDCOPY
GO TO L8
L6:  PRINT KA,VA
L8:  PRINT "END OF PROGRAM"
```

```
EDITING COMATRCS
PROGRAM
$
$ THIS PROGRAM PROVIDES FOR THE CALCULATION OF THE COEFFICIENT
$MATRICES THAT ARE NEEDED IN THE SECOND ORDER CALCULATIONS.
$SINCE THESE MATRICES ARE FUNCTIONS ONLY OF THE SYSTEM DESCRIPTION
$AND PERMUTATION MATRICES, THEY MAY BE TREATED AS CONSTANTS IN THE
$EXECUTION OF THE SECORDA PROGRAM, WHICH FOLLOWS.  THESE COEFFICIENT
$MATRICES ARE CALLED CM1,CM2,CM3,CM4,CM5,AND CM6.  THESE MATRICES
$ARE THEN STORED FOR FUTURE CALCULATION.  THE MELD OPERATION USED IN
$THIS ROUTINE GIVES AN ODOMETER ORDERING OF ITS ARGUMENTS, WHICH ARE
$ARRAYS, BY ALTERING THEIR STRUCTURE.  FOR A COMPLETE DESCRIPTION OF
$THIS OPERATION, PLEASE REFER TO REFERENCE [23].
$
$INITIALIZE PERMUTATION AND SYSTEM MATRICES
$
S121=MAT(8,8')
S221=MAT(8,8')
S111=MAT(4,4'); S112=MAT(8,8')
A01=MAT(2,2'); A10=MAT(2,2'); A11=MAT(2,4')
A02=MAT(2,4'); A20=MAT(2,4')
$
$RETRIEVE DATA FILES AND LOAD MATRICES
$
GET A01MAT; GET A10MAT; GET A11MAT; GET A02MAT; GET A20MAT
GET S111MAT; GET S121MAT; GET S221MAT; GET S112MAT
LOADDATA(S111,S11MAT);LOADDATA(S121,S121MAT);LOADDATA(S221,S221MAT)
LOADDATA(S112,S112MAT); LOADDATA(A01,A01MAT); LOADDATA(A10,A10MAT)
LOADDATA(A11,A11MAT); LOADDATA(A02,A02MAT); LOADDATA(A20,A20MAT)
P1=MAT(4,8'); P2=MAT(4,8'); P4=MAT(4,4'); P5=MAT(4,4')
P7=MAT(8,8'); P9=MAT(8,8'); P11=MAT(8,8')
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
T=MAT(4,4)
$
$CALCULATE CM1 MATRIX
$
J1=1,2, J2=1,2,3,4, J3=1,3, J4=1,3,5,7
MELD(J1,J2), MELD(J3,J4)
FOR I=1,8
P1(J3(I),J4(I))=A11(J1(I),J2(I))*UMAT(2)
ENDLOOP I
P3=P1*S121
P2(1,1)=A11
P2(3,5)=A11
CM1=P2+P3
KEEP CM1
$
$CALCULATE CM2 MATRIX
$
FOR I=1,8
P1(J3(I),J4(I))=A02(J1(I),J2(I))*UMAT(2)
ENDLOOP I
P3=P1*S112
P2(1,1)=A02
P2(3,5)=A02
CM2=P2+P3
KEEP CM2
$
$CALCULATE CM6 MATRIX
$
FOR I=1,8
P1(J3(I),J4(I))=A20(J1(I),J2(I))*UMAT(2)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ENDLOOP I
P2(1,1)=A20
P2(3,5)=A20
CM6=P1+P2
KEEP CM6
$
$CALCULATE CM5 MATRIX
$
J1=1,2; J5=J1; J3=1,3; J6=J3
MELD(J1,J5); MELD(J3,J6)
FOR I=1,4
P4(J3(I),J6(I))=A01(J1(I),J5(I))*UMAT(2)
ENDLOOP I
P5(1,1)=A01
P5(3,3)=A01
P6=P4*S111
CM5=P6+P5
KEEP CM5
$
$CALCULATE CM3 MATRIX
$
J7=1,5; J8=J7
MELD(J7,J8)
FOR I=1,4
P7(J7(I),J8(I))=A01(J1(I),J5(I))*UMAT(4)
ENDLOOP I
P8=P7*S221
FOR I=1,4
T(J3(I),J6(I))=A01(J1(I),J5(I))*UMAT(2)
ENDLOOP I
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
P9(1,1)=T; P9(5,5)=T
P10=P9*S121
FOR J=1,7,2
P11(J,J)=A01
ENDLOOP J
CM3=P8+P10+P11
KEEP CM3
$
$CALCULATE CM4 MATRIX
$
FOR I=1,4
P7(J7(I),J8(I))=A10(J1(I),J5(I))*UMAT(4)
ENDLOOP I
FOR I=1,4
T(J3(I),J6(I))=A10(J1(I),J5(I))*UMAT(2)
ENDLOOP I
P9(1,1)=T; P9(5,5)=T
FOR J=1,7,2
P11(J,J)=A10
ENDLOOP J
CM4=P7+P9+P11
KEEP CM4
```

ORIGINAL PAGE IS
OF POOR QUALITY

EDITING SECORDA

```
PROGRAM
$
$ THIS PROGRAM PROVIDES THE SECOND ORDER ANALYSIS OF THE EXAMPLE
$PROBLEM, WHICH IS THE CALCULATION OF V3(T) AND K2(T), WITH THE
$LATER SUBSTITUTION AND INTEGRATION TO GET THE CONTROLLED SYSTEM
$RESPONSE.  THE FIRST PART INTEGRATES THE DIFFERENTIAL
$EQUATION FROM THE BOUNDARY CONDITION OF V3(T1)=M3=0 TO AN INITIAL
$VALUE OF V3(0), WHICH IS FREE.
$
$RETRIEVE COEFFICIENT MATRICES CM1 TO CM6.  THESE WERE CALCULATED
$IN THE COMATRCS ROUTINE.  ALSO, RETRIEVE DATA FILES FPR PERFORMANCE
$INDEX MATRICES, INITIALIZE, AND LOAD
$
GET CM1; GET CM2; GET CM3; GET CM4; GET CM5; GET CM6
GET CMAT; GET RMAT; GET QMAT; GET KA; GET VA; GET M2VEC
Q11=VECTOR(4:); G=MAT(8,8:)
Q02=VECTOR(4:); M2=MAT(2,2:)
Q20=VECTOR(4:); LOADDATA(M2,M2VEC)
LOADDATA(Q11,CMAT); LOADDATA(Q02,RMAT); LOADDATA(Q20,QMAT)
Q=MAT(2,2:Q20(1),Q20(2),Q20(3),Q20(4))
C=MAT(2,2:Q11(1),Q11(2),Q11(3),Q11(4))
R=MAT(2,2:Q02(1),Q02(2),Q02(3),Q02(4))
$
$NOW, CALCULATE THE G MATRIX, WHICH WILL BE USED TO EXPRESS
$K2 IN TERMS OF V3
$
G1=MAT(1,2:2*Q02(1),2*Q02(2))
G2=MAT(1,2:2*Q02(2),2*Q02(4))
J=1
FOR I=1,4
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
G(I,J)=G1
J=J+2
ENDLOOP I
J=1
FOR I=5,8
G(I,J)=G2
J=J+2
ENDLOOP I
$
$INITIALIZE ARRAYS
$
REQUEST T1,DT
S=(T1/(-DT))+1
X=ARRAY(2:); U=ARRAY(2:); K2A=ARRAY(S,9:)
V3A=ARRAY(S,9:); M3=VECTOR(8:)
E1=MAT(8,8:); E2=MAT(4,8:); E3=MAT(8,8:)
E4=MAT(4,8:); E5=MAT(4,8:); SM=MAT(4,4:)
$
$INITIALIZE V3, POINTER R1
$
V3=M3; V3ARY=AFAM(V3)
R1=S
V2=VECTOR(4:)
VM1=MAT(8,8:); B=MAT(4,4:)
FOR T=T1,0,DT
$
$PICK OUT THE V2 VALUES FROM THE FUNCTION ARRAY AND LOAD A
$4-VECTOR CALLED V2
$
V2(1)=VA(R1,2); V2(2)=VA(R1,3)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
V2(3)=VA(R1,4); V2(4)=VA(R1,5)
$
$UPDATE VM1
$
B(1,)=KA(R1,2),0,KA(R1,3),0
B(2,)=.5*KA(R1,4),.5*KA(R1,2),.5*KA(R1,5),.5*KA(R1,3)
B(3,)=B(2,)
B(4,)=0,KA(R1,4),0,KA(R1,5)
VM1(1,1)=B; VM1(5,5)=B
$
$CALCULATE K2, FORM MATRICES, LOAD ARRAYS
$
K2=-(V2*(CM1+2*CM2*VM1)+V3*CM3)*(1/G)
K2ARY=AFAM(K2), V3ARY=AFAM(V3)
K2A(R1,1)=T; K2A(R1,2)=K2ARY
V3A(R1,1)=T; V3A(R1,2)=V3ARY
K1M=MAT(2,2·KA(R1,2),KA(R1,3),KA(R1,4),KA(R1,5))
K2M=MAT(2,4·)
K2M(1,)=K2(1),K2(2),K2(3),K2(4)
K2M(2,)=K2(5),K2(6),K2(7),K2(8)
$
$UPDATE E1, E2, E3, E4, AND E5, WHICH ARE KRONECKER PRODUCT MATRICES
$AND ARE FUNCTIONS OF K1 AND K2.  HERE, WE USE THE MELD OPERATION
$THAT IS DESCRIBED IN REFERENCE [23].
$
FOR J=1,7,2
E1(J,J)=K1M
ENDLOOP J
E2(1,1)=K2M; E2(3,5)=K2M
J1=1,2; J2=J1; J3=1,3; J4=J3
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
MELD(J1,J2); MELD(J3,J4)
FOR I=1,4
SM(J3(I),J4(I))=K1M(J1(I),J2(I))*K1M
ENDLOOP I
E3(1,1)=SM; E3(5,5)=SM
J5=1,3; J6=1,5
MELD(J5,J6)
FOR I=1,4
E4(J5(I),J6(I))=K1M(J1(I),J2(I))*K2M
ENDLOOP I
J7=1,2; J8=1,2,3,4; J9=1,3; J10=1,3,5,7
MELD(J7,J8); MELD(J9,J10)
FOR I=1,8
E5(J9(I),J10(I))=K2M(J7(I),J8(I))*K1M
ENDLOOP I
$
$UPDATE V3
$
V3MP=V3+(DT/2)*(-V3*(CM3*E1+CM4)-V2*(CM5*E2+CM1*E1+CM2*E3+CM6)
&-Q02*(E4+E5)-Q11*E2)
V3=V3+DT*(-V3MP*(CM3*E1+CM4)-V2*(CM5*E2+CM1*E1+CM2*E3+CM6)
&-Q02*(E4+E5)-Q11*E2)
$
$UPDATE THE R1 INDEX AND T TO CONTINUE INTEGRATION
$
R1=R1-1
ENDLOOP T
PRINT "ENTER CHOICE=1 TO CONTINUE OR CHOICE=0 TO STOP AND
&DISPLAY K2A AND V3A"
REQUEST CHOICE
```

ORIGINAL PAGE IS OF POOR QUALITY

```
IF(CHOICE.EQ.0) GO TO L6
$
$ THE NEXT SECTION PERFORMS THE NECESSARY INTEGRATIONS
$AFTER THE SUBSTITUTION OF THE OPTIMAL CONTROL U(T)IS MADE INTO THE
$SYSTEM EQUATIONS.  THE METHOD OF NUMERICAL INTEGRATION USED IS A
$FOURTH ORDER RUNGE-KUTTA ROUTINE USING KUTTA'S COEFFICIENTS.  THIS
$PART OF THE PROGRAM PRODUCES THE ARRAYS X1A AND X2A, WHICH CONTAIN
$THE STATE VARIABLES AND THE ARRAYS U1A AND U2A, WHICH CONTAIN
$THE CONTROLLER COMPONENTS, AGAIN SEQUENCED BY TIME.  THESE ARRAYS
$WILL BE USED LATER TO GENERATE THE NECESSARY TIME PLOTS.
$
PRINT "OPTIMAL VALUE INTEGRATION COMPLETE"
$
$INITIALIZE MAIN COUNT (MCT), RUNGE KUTTA COEFFICIENT ARRAYS
$RKC1 AND RKC2, FUNCTION ARRAYS X1A, X2A, U1A, U2A, AND TIME
$
RUNGE:  MCT=1; PERINDX=0
REQUEST X10,X20
H=-DT;RKC1=ARRAY(5:);RKC2=ARRAY(5:)
X1A=ARRAY(5:); X2A=X1A; U1A=X1A; U2A=X1A
TA=X1A
X(1)=X10;X(2)=X20;T=0
L5:  XP=X; COUNT=1
$
$UPDATE CONTROL FUNCTION AND SUBSTITUTE INTO SYSTEM EQUATIONS
$
L2:  U11=KA(MCT,2)*XP(1)+KA(MCT,3)*XP(2)
U12=K2A(MCT,2)*XP(1)*XP(1)+K2A(MCT,3)*XP(1)*XP(2)+K2A(MCT,4)
&*XP(2)*XP(1)+K2A(MCT,5)*XP(2)*XP(2)
U(1)=U11+U12
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
U21=KA(MCT,4)*XP(1)+KA(MCT,5)*XP(2)
U22=K2A(MCT,6)*XP(1)*XP(1)+K2A(MCT,7)*XP(1)*XP(2)+K2A(MCT,8)
&*XP(2)*XP(1)+K2A(MCT,9)*XP(2)*XP(2)
U(2)=U21+U22
F1A=U(2)*(EXP(XP(1)*XP(2))+EXP(-XP(1)*XP(2)))*.5-EXP(2*U(1))
&*(EXP(2*XP(1))-EXP(-2*XP(1)))*.5
F1B=-3*(EXP(XP(2))-EXP(-XP(2)))*.5
F1=F1A+F1B
F2A=EXP(U(1)*U(2))*(EXP(XP(1))-EXP(-XP(1)))*.5-EXP(U(1))*U(1)*
&(EXP(XP(1)*XP(1))+EXP(-XP(1)*XP(1)))*.5
F2B=(EXP(XP(2))-EXP(-XP(2)))*.5
F2=F2A+F2B
$
$CALCULATE RUNGE KUTTA COEFFICIENTS
$
RKC1(COUNT)=F1*H, RKC2(COUNT)=F2*H
IF(COUNT.EQ.4) GO TO L4
$
$UPDATE PROVISIONAL X FOR NEXT COEFFICIENT
$
XP(1)=X(1)+.5*RKC1(COUNT)
XP(2)=X(2)+.5*RKC2(COUNT)
COUNT=COUNT+1
IF(COUNT.EQ.4) GO TO L3
GO TO L2
L3,  XP(1)=X(1)+RKC1(3)
XP(2)=X(2)+RKC2(3)
GO TO L2
$
$CALCULATE DELTA X
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
$
L4·   RKC1(5)=(RKC1(1)+2*RKC1(2)+2*RKC1(3)+RKC1(4))/6
RKC2(5)=(RKC2(1)+2*RKC2(2)+2*RKC2(3)+RKC2(4))/6
$
$UPDATE FUNCTION ARRAYS U1A, U2A, X1A, AND X2A
$
X1A(MCT)=X(1), X2A(MCT)=X(2)
U1A1=KA(MCT,2)*X(1)+KA(MCT,3)*X(2)
U1A2=K2A(MCT,2)*X(1)*X(1)+K2A(MCT,3)*X(1)*X(2)+K2A(MCT,4)
&*X(2)*X(1)+K2A(MCT,5)*X(2)*X(2)
U1A(MCT)=U1A1+U1A2
U2A1=KA(MCT,4)*X(1)+KA(MCT,5)*X(2)
U2A2=K2A(MCT,6)*X(1)*X(1)+K2A(MCT,7)*X(1)*X(2)+K2A(MCT,8)
&*X(2)*X(1)+K2A(MCT,9)*X(2)*X(2)
U2A(MCT)=U2A1+U2A2
TA(MCT)=T
$
$UPDATE PERFORMANCE INDEX
$
UV=VECTOR(2:U1A(MCT),U2A(MCT)), XV=VFAM(X)
PERINDX=PERINDX+(XV*Q*XV+UV*R*UV+XV*C*UV)
$
$UPDATE STATE VARIABLES FOR NEXT GO AROUND
$
X(1)=X(1)+RKC1(5)
X(2)=X(2)+RKC2(5)
$
$UPDATE MCT, TIME FOR NEXT GO AROUND
$
MCT=MCT+1, T=T+H
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS POOR QUALITY

```
IF(MCT.NE.S+1) GO TO L5
PERINDX=.5*(XU*M2*XU+PERINDX)
PRINT PERINDX
PRINT "STATE INTEGRATION COMPLETE"
$
$ WE ARE NOW READY TO BEGIN THE MECHANICS OF PLOTTING THE STATE
$AND CONTROL VARIABLES VERSUS TIME.  BECAUSE OF THE LARGE SIZES OF
$THE ARRAYS, IT WILL BE USEFUL IF WE SELECT ONLY A CERTAIN NUMBER
$OF POINTS FROM EACH ARRAY THAT ACTUALLY WILL BE PLOTTED.  THIS WILL
$BE DONE BY SPECIFYING THE PARAMETER SEL.  ONCE THESE SMALLER ARRAYS
$HAVE BEEN FORMED, THE FINAL STEP IN THE PROCEDURE IS THE PLOTTING
$OF FOUR GRAPHS - X(1), X(2), U(1), AND U(2) ALL VERSUS TIME
$OVER THE INTERVAL, WHICH IS USER SPECIFIED FROM T=0 TO T=T1.
$
GRONLY.  REQUEST SEL
FU=1+(SEL)*INTPART(S/SEL)
TAPL=ARRAY(1:)
X1APL=TAPL, X2APL=TAPL, U1APL=TAPL, U2APL=TAPL
INDX=1
$
$FORM REDUCED SIZE ARRAYS FOR PLOTTING
$
IF(FRAC(S/SEL).EQ.0) FU=(S+1)-SEL
FOR J=1,FU,SEL
TAPL(INDX)=TA(J)
X1APL(INDX)=X1A(J)
X2APL(INDX)=X2A(J)
U1APL(INDX)=U1A(J)
U2APL(INDX)=U2A(J)
INDX=INDX+1
```

```
ENDLOOP J
$
$PLOTTING BEGINS HERE
$
GRAPHICS(TEK4010)
SETXLABEL 'TIME IN SECONDS'
SETYLABEL 'X1'
SETTITLE 'STATE VARIABLE X1 VS. TIME'
LINECODE=1
GRAPH(X1APL,TAPL)
HARDCOPY
SETYLABEL 'X2'
SETTITLE 'STATE VARIABLE X2 VS. TIME'
GRAPH(X2APL,TAPL)
HARDCOPY
SETYLABEL 'U1'
SETTITLE 'CONTROL VARIABLE U1 VS. TIME'
GRAPH(U1APL,TAPL)
HARDCOPY
SETYLABEL 'U2'
SETTITLE'CONTROL VARIABLE U2 VS. TIME'
GRAPH(U2APL,TAPL)
HARDCOPY
GO TO L8
L6:  PRINT K2A,V3A
L8:  PRINT "END OF PROGRA
```

ORIGINAL PAGE IS
OF POOR QUALITY

KA(INTEGERS(1,35),) (A 35 BY 5 ARRAY)

0	-.56702	.10455	-.036163	-.63296
.05	-.567	.10451	-.036242	-.63298
.1	-.56695	.10447	-.036335	-.63302
.15	-.56689	.10446	-.036439	-.63308
.2	-.5668	.10448	-.036554	-.63317
.25	-.56669	.10453	-.036677	-.63328
.3	-.56655	.10462	-.036807	-.63342
.35	-.56639	.10475	-.03694	-.63358
.4	-.56621	.10493	-.037073	-.63377
.45	-.566	.10517	-.037202	-.63397
.5	-.56579	.10546	-.037324	-.63419
.55	-.56556	.1058	-.037435	-.63441
.6	-.56533	.10619	-.037532	-.63464
.65	-.5651	.10663	-.03761	-.63487
.7	-.56489	.1071	-.037667	-.63508
.75	-.56469	.10761	-.037701	-.63528
.8	-.56452	.10814	-.03771	-.63545
.85	-.56438	.10868	-.037695	-.63558
.9	-.56429	.10921	-.037655	-.63568
.95	-.56424	.10972	-.037594	-.63573
1	-.56424	.11019	-.037514	-.63573
1.05	-.5643	.11062	-.03742	-.63567
1.1	-.5644	.11097	-.037319	-.63556
1.15	-.56456	.11125	-.037218	-.6354
1.2	-.56477	.11145	-.037126	-.6352
1.25	-.56502	.11154	-.037052	-.63495
1.3	-.5653	.11153	-.037008	-.63467
1.35	-.56559	.11143	-.037004	-.63438
1.4	-.56589	.11123	-.03705	-.63408
1.45	-.56617	.11094	-.037157	-.63381
1.5	-.56641	.11059	-.037335	-.63356
1.55	-.5666	.11019	-.037591	-.63338
1.6	-.56671	.10977	-.037932	-.63327
1.65	-.56671	.10937	-.038361	-.63326
1.7	-.5666	.10902	-.038879	-.63337

ORIGINAL PAGE IS OF POOR QUALITY

KA(INTEGERS(36,70),) (A 35 BY 5 ARRAY)

1.75	-.56636	.10876	-.039485	-.63362
1.8	-.56596	.10864	-.040173	-.63402
1.85	-.56539	.10869	-.040933	-.63459
1.9	-.56466	.10897	-.041755	-.63532
1.95	-.56376	.10951	-.04262	-.63622
2	-.56269	.11035	-.043512	-.63729
2.05	-.56147	.11152	-.044407	-.63851
2.1	-.56012	.11304	-.045283	-.63985
2.15	-.55866	.11492	-.046114	-.64131
2.2	-.55713	.11717	-.046875	-.64283
2.25	-.55557	.11976	-.04754	-.64439
2.3	-.55402	.12268	-.048086	-.64593
2.35	-.55254	.12589	-.048493	-.64741
2.4	-.55117	.12933	-.048745	-.64878
2.45	-.54997	.13293	-.04883	-.64998
2.5	-.54899	.13661	-.048743	-.65095
2.55	-.54828	.14028	-.048489	-.65166
2.6	-.54788	.14384	-.048078	-.65206
2.65	-.54783	.14718	-.047532	-.65211
2.7	-.54815	.1502	-.046881	-.65179
2.75	-.54884	.15279	-.046166	-.6511
2.8	-.5499	.15485	-.045438	-.65004
2.85	-.55131	.15631	-.044755	-.64863
2.9	-.55302	.1571	-.044185	-.64693
2.95	-.55496	.15717	-.043801	-.64499
3	-.55705	.15653	-.043679	-.64291
3.05	-.55919	.15519	-.043898	-.64078
3.1	-.56124	.15322	-.044532	-.63873
3.15	-.56308	.15071	-.045652	-.6369
3.2	-.56455	.14783	-.047321	-.63543
3.25	-.56551	.14476	-.049585	-.63449
3.3	-.56578	.14172	-.05248	-.63422
3.35	-.56523	.13897	-.056018	-.63478
3.4	-.56372	.13681	-.060193	-.6363
3.45	-.56112	.13555	-.064975	-.63889

ORIGINAL PAGE IS
OF POOR QUALITY

KA(INTEGERS(71,101),) (A 31 BY 5 ARRAY)

3.5	-.55735	.13551	-.07031	-.64266
3.55	-.55235	.13701	-.076121	-.64766
3.6	-.54609	.14037	-.082308	-.65391
3.65	-.53858	.14587	-.088747	-.66141
3.7	-.52989	.15378	-.0953	-.67009
3.75	-.52012	.16431	-.10181	-.67985
3.8	-.50941	.17762	-.10811	-.69053
3.85	-.49798	.19383	-.11402	-.70195
3.9	-.48605	.21295	-.11937	-.71385
3.95	-.47392	.23493	-.12398	-.72595
4	-.46192	.25963	-.12769	-.73792
4.05	-.45041	.28682	-.13034	-.7494
4.1	-.43978	.31615	-.13182	-.76
4.15	-.43046	.34718	-.13203	-.7693
4.2	-.42287	.37933	-.13091	-.77686
4.25	-.41746	.41192	-.12847	-.78225
4.3	-.41463	.44416	-.12478	-.78507
4.35	-.41477	.47514	-.11996	-.78491
4.4	-.4182	.50384	-.11427	-.78148
4.45	-.42515	.52921	-.10801	-.77454
4.5	-.43571	.55014	-.10166	-.764
4.55	-.4498	.56555	-.09576	-.74994
4.6	-.46715	.57446	-.091008	-.73264
4.65	-.4872	.57607	-.088196	-.71264
4.7	-.50916	.56987	-.088212	-.69075
4.75	-.5319	.55574	-.092005	-.66808
4.8	-.55404	.53405	-.10054	-.64603
4.85	-.57389	.50573	-.11476	-.62626
4.9	-.58959	.47237	-.13552	-.61065
4.95	-.59912	.43619	-.16354	-.6012
5	-.6004	.4	-.1994	-.6

ORIGINAL PAGE IS
OF POOR QUALITY

K2A(INTEGERS(1,35),) (A 35 BY 9 ARRAY)

0	.063221	.031509	-.097294	-.17558	.0014166	.044615	.04598	.040294
.05	.063455	.031622	-.097101	-.17524	.0014606	.044696	.046051	.040225
.1	.063683	.031728	-.096936	-.1749	.0015315	.044811	.046159	.040213
.15	.063897	.031823	-.096814	-.17459	.0016307	.044963	.04631	.040272
.2	.064087	.031899	-.096751	-.17435	.0017591	.045153	.046504	.04041
.25	.064243	.031952	-.096762	-.1742	.0019163	.04538	.046744	.040636
.3	.064357	.031975	-.096861	-.17418	.0021009	.045644	.047027	.040956
.35	.064421	.031965	-.097061	-.1743	.0023102	.045942	.047352	.041373
.4	.064427	.031917	-.097371	-.17461	.0025404	.046269	.047714	.041885
.45	.064369	.031829	-.0978	-.17513	.0027868	.046619	.048106	.042487
.5	.064244	.031701	-.098349	-.17586	.0030431	.046985	.048522	.043171
.55	.064052	.031532	-.099018	-.17684	.0033026	.04736	.04895	.043924
.6	.063792	.031325	-.099801	-.17806	.0035574	.047734	.04938	.044728
.65	.06347	.031084	-.10069	-.17951	.0037996	.048098	.049801	.045563
.7	.063092	.030813	-.10166	-.18119	.0040204	.048441	.0502	.046405
.75	.062669	.030519	-.10271	-.18308	.0042117	.048754	.050564	.047227
.8	.062214	.030209	-.1038	-.18513	.0043652	.049028	.050882	.048002
.85	.061744	.029894	-.10491	-.18732	.0044736	.049255	.05114	.048699
.9	.061278	.029583	-.106	-.18958	.0045309	.049429	.05133	.049293
.95	.060838	.029287	-.10704	-.19187	.0045323	.049544	.051444	.049754
1	.060446	.029016	-.108	-.19411	.004475	.049599	.051475	.050061
1.05	.060128	.028781	-.10884	-.19625	.0043585	.049594	.051421	.050192
1.1	.059908	.028592	-.10953	-.19821	.0041849	.049533	.051282	.050135
1.15	.059811	.02846	-.11004	-.19992	.0039593	.049422	.051063	.049882
1.2	.05986	.028392	-.11033	-.20133	.0036898	.049271	.050773	.049434
1.25	.060074	.028396	-.11039	-.20237	.0033882	.049093	.050424	.0488
1.3	.060469	.028476	-.1102	-.20301	.0030693	.048906	.050034	.047997
1.35	.061055	.028634	-.10977	-.2032	.0027518	.04873	.049624	.047055
1.4	.061836	.02887	-.10909	-.20293	.0024573	.048589	.049222	.046012
1.45	.062807	.029178	-.10817	-.20221	.0022103	.048508	.048856	.044916
1.5	.063953	.029549	-.10707	-.20106	.0020377	.048516	.048562	.043823
1.55	.065253	.029969	-.1058	-.19953	.0019683	.048645	.048374	.0428
1.6	.066673	.030419	-.10445	-.19769	.0020312	.048925	.048333	.041919
1.65	.068169	.030874	-.10306	-.19564	.0022556	.049388	.048478	.041258
1.7	.069688	.031304	-.10173	-.19352	.0026689	.050063	.048849	.040899

ORIGINAL PAGE IS OF POOR QUALITY

K2A(INTEGERS(36,70),) (A 35 BY 9 ARRAY)

1.75	.071168	.031674	-.10056	-.19146	.0032955	.050981	.049483	.040924
1.8	.072539	.031941	-.099645	-.18965	.0041558	.052164	.050415	.041414
1.85	.073729	.032062	-.099097	-.18827	.0052638	.053635	.051675	.042447
1.9	.074659	.031988	-.099032	-.18754	.0066266	.055405	.053287	.044089
1.95	.075254	.03167	-.099563	-.18767	.0082424	.057481	.055263	.0464
2	.075442	.031058	-.1008	-.18889	.0101	.059859	.057608	.049422
2.05	.075157	.030104	-.10283	-.1914	.012176	.062527	.060315	.053179
2.1	.074346	.028766	-.10575	-.19543	.014439	.065458	.063361	.057675
2.15	.072972	.027009	-.10962	-.20114	.016843	.068615	.06671	.062889
2.2	.071014	.024807	-.11447	-.2087	.019333	.071947	.070311	.06877
2.25	.068475	.022148	-.1203	-.21823	.021844	.075392	.074096	.075241
2.3	.065385	.019037	-.12708	-.22978	.024302	.078875	.077985	.082191
2.35	.061799	.015497	-.13474	-.24337	.026628	.082312	.081881	.089475
2.4	.057802	.011571	-.14317	-.25895	.028738	.085607	.085677	.096921
2.45	.053507	.0073292	-.1522	-.27637	.030549	.088664	.08926	.10432
2.5	.049057	.0028627	-.16164	-.29543	.031982	.091381	.092509	.11145
2.55	.04462	-.0017099	-.17122	-.31583	.032965	.093661	.C95308	.11804
2.6	.040389	-.0062461	-.18067	-.33719	.033438	.095418	.097547	.12385
2.65	.036568	-.010582	-.18967	-.35902	.033359	.096577	.09913	.12859
2.7	.033374	-.014536	-.19787	-.38077	.032706	.097088	.099986	.13201
2.75	.031023	-.017918	-.20492	-.40182	.031481	.096928	.10007	.13388
2.8	.029719	-.020534	-.21048	-.4215	.029714	.09611	.099386	.13401
2.85	.029644	-.022202	-.21423	-.43911	.027463	.094685	.097967	.13225
2.9	.030944	-.022762	-.21592	-.45394	.024814	.09275	.095907	.12858
2.95	.033723	-.022094	-.21535	-.46535	.021878	.090445	.093349	.12301
3	.038027	-.020125	-.21242	-.47276	.018788	.087953	.090487	.11571
3.05	.043844	-.016847	-.20715	-.47574	.015691	.085493	.087558	.10694
3.1	.051098	-.012324	-.1997	-.47404	.012737	.083313	.084838	.097092
3.15	.059653	-.0066909	[illegible]	-.46763	.010075	.081673	.082622	.086665
3.2	.06932	-1.5465E-4	[illegible]	-.45674	.0078339	.080827	.081207	.076256
3.25	.079873	.0070218	[illegible]	-.44188	.0061186	.081007	.080871	.066534
3.3	.09106	.014539	-.15579	-.42382	.0049958	.0824	.081849	.0582
3.35	.10263	.022089	-.14419	-.40362	.0044882	.085128	.084311	.051951
3.4	.11434	.029392	-.13372	-.38252	.0045697	.089232	.088339	.048424
3.45	.12601	.036232	-.12507	-.36193	.0051645	.094656	.093914	.048157

ORIGINAL PAGE IS
OF POOR QUALITY

K2A(INTEGERS(71,101),) (A 31 BY 9 ARRAY)

3.5	.13748	.042495	-.11882	-.34332	.0061502	.10124	.1009	.051532
3.55	.14867	.048202	-.1154	-.32812	.0073649	.10873	.10906	.058743
3.6	.15957	.053537	-.11506	-.31763	.0086169	.11675	.11803	.069757
3.65	.17022	.058861	-.11784	-.31287	.0096982	.12488	.12736	.084297
3.7	.18071	.064715	-.12353	-.31454	.010399	.13259	.1365	.10183
3.75	.19119	.071808	-.13166	-.32288	.010522	.13935	.14487	.12156
3.8	.20176	.080987	-.14156	-.33761	.0099021	.14459	.15186	.14248
3.85	.21254	.093198	-.15232	-.35789	.0084177	.14778	.15687	.16334
3.9	.22358	.10942	-.16291	-.38227	.0060073	.14844	.15935	.18272
3.95	.23481	.13059	-.1722	-.40871	.0026808	.1462	.15884	.1991
4	.24605	.15753	-.17906	-.43461	-.0014716	.14081	.15503	.21088
4.05	.25698	.19079	-.18247	-.45686	-.0062751	.13224	.14779	.2165
4.1	.26707	.2306	-.18163	-.472	-.011474	.12065	.13722	.21451
4.15	.27564	.27665	-.17608	-.47635	-.016746	.10648	.12372	.20373
4.2	.28183	.32804	-.16586	-.46627	-.021725	.090475	.10796	.18339
4.25	.2847	.38308	-.15166	-.43848	-.026049	.073661	.09098	.15329
4.3	.28327	.43926	-.13491	-.39046	-.029415	.057339	.074079	.11398
4.35	.27677	.49315	-.11784	-.32089	-.031658	.043005	.058794	.066885
4.4	.26479	.54053	-.10346	-.23017	-.032831	.032205	.04674	.014465
4.45	.24757	.57656	-.095383	-.12081	-.033283	.026334	.039399	-.039858
4.5	.22637	.59621	-.097468	.002158	-.03369	.026353	.037828	-.09184
4.55	.20373	.59493	-.11323	.13125	-.035026	.032487	.042343	-.1368
4.6	.1837	.56941	-.14509	.25685	-.038411	.043937	.052206	-.17021
4.65	.17185	.51857	-.19352	.36813	-.044813	.058699	.065442	-.1885
4.7	.17498	.44441	-.2562	.45454	-.054586	.073611	.078877	-.18991
4.75	.20046	.3526	-.32758	.50761	-.066872	.084708	.088517	-.17516
4.8	.25507	.25254	-.39888	.52291	-.078962	.087951	.090308	-.14787
4.85	.34368	.15658	-.45902	.50156	-.08572	.080285	.081266	-.11421
4.9	.46785	.078566	-.49616	.45075	-.079227	.060883	.060789	-.081896
4.95	.62461	.031697	-.50004	.38312	-.048748	.032357	.031952	-.058536
5	.80586	.02624	-.46428	.31464	.018999	.00172	.0025286	-.04944

ORIGINAL PAGE IS
OF POOR QUALITY

VA(INTEGERS(1,35),) (A 35 BY 5 ARRAY)

0	.18136	.16529	.16529	.5231
.05	.18176	.16539	.16539	.52287
.1	.18222	.16559	.16559	.5227
.15	.18274	.16591	.16591	.52264
.2	.18332	.16635	.16635	.52272
.25	.18394	.16692	.16692	.52297
.3	.18459	.16761	.16761	.52342
.35	.18526	.16842	.16842	.52409
.4	.18592	.16934	.16934	.52501
.45	.18657	.17036	.17036	.52618
.5	.18718	.17145	.17145	.52763
.55	.18774	.17258	.17258	.52933
.6	.18822	.17373	.17373	.53129
.65	.18862	.17486	.17486	.53348
.7	.1889	.17594	.17594	.53587
.75	.18907	.17692	.17692	.53841
.8	.18912	.17777	.17777	.54106
.85	.18904	.17845	.17845	.54374
.9	.18884	.17893	.17893	.54639
.95	.18853	.17918	.17918	.54895
1	.18813	.17917	.17917	.55132
1.05	.18766	.17889	.17889	.55344
1.1	.18715	.17835	.17835	.55523
1.15	.18665	.17755	.17755	.55663
1.2	.18619	.17651	.17651	.55758
1.25	.18582	.17527	.17527	.55805
1.3	.1856	.17388	.17388	.55801
1.35	.18557	.17241	.17241	.55748
1.4	.18581	.17093	.17093	.55647
1.45	.18635	.16954	.16954	.55504
1.5	.18724	.16832	.16832	.55327
1.55	.18852	.16739	.16739	.55127
1.6	.19023	.16685	.16685	.54919
1.65	.19238	.16681	.16681	.54718
1.7	.19498	.16737	.16737	.54542

ORIGINAL PAGE IS
OF POOR QUALITY

VA(INTEGERS(36,70),) (A 35 BY 5 ARRAY)

1.75	.19802	.16861	.16861	.54413
1.8	.20147	.17062	.17062	.54352
1.85	.20528	.17345	.17345	.54381
1.9	.2094	.17712	.17712	.54521
1.95	.21374	.18165	.18165	.54792
2	.21821	.187	.187	.55213
2.05	.22271	.19311	.19311	.55799
2.1	.2271	.19987	.19987	.5656
2.15	.23126	.20717	.20717	.57502
2.2	.23508	.21482	.21482	.58626
2.25	.23841	.22263	.22263	.59926
2.3	.24115	.23036	.23036	.61388
2.35	.2432	.23779	.23779	.62992
2.4	.24446	.24463	.24463	.64712
2.45	.24488	.25063	.25063	.66513
2.5	.24445	.25552	.25552	.68354
2.55	.24317	.25907	.25907	.7019
2.6	.24111	.26106	.26106	.7197
2.65	.23837	.26132	.26132	.73641
2.7	.23511	.25974	.25974	.7515
2.75	.23152	.25627	.25627	.76445
2.8	.22787	.25095	.25095	.77477
2.85	.22445	.24391	.24391	.78205
2.9	.22159	.23536	.23536	.78597
2.95	.21966	.22564	.22564	.78633
3	.21905	.21518	.21518	.78308
3.05	.22015	.20451	.20451	.77635
3.1	.22333	.19423	.19423	.76646
3.15	.22895	.18505	.18505	.75394
3.2	.23732	.17771	.17771	.73951
3.25	.24867	.17297	.17297	.72413
3.3	.26319	.17162	.17162	.70892
3.35	.28093	.17441	.17441	.6952
3.4	.30187	.18202	.18202	.68443
3.45	.32585	.19504	.19504	.67815

ORIGINAL PAGE IS OF POOR QUALITY

VA(INTEGERS(71,101),) (A 31 BY 5 ARRAY)

3.5	.35261	.21394	.21394	.67798
3.55	.38175	.23901	.23901	.68553
3.6	.41278	.27038	.27038	.70237
3.65	.44507	.30798	.30798	.72995
3.7	.47793	.35151	.35151	.76958
3.75	.51058	.40044	.40044	.82233
3.8	.54216	.45402	.45402	.88903
3.85	.57181	.51126	.51126	.97016
3.9	.59864	.57094	.57094	1.0659
3.95	.62176	.63163	.63163	1.1759
4	.64035	.69167	.69167	1.2996
4.05	.65367	.74926	.74926	1.4356
4.1	.66109	.80241	.80241	1.5824
4.15	.66213	.84903	.84903	1.7376
4.2	.65653	.88695	.88695	1.8984
4.25	.64431	.91401	.91401	2.0615
4.3	.62577	.92811	.92811	2.2227
4.35	.60163	.92735	.92735	2.3775
4.4	.57305	.91013	.91013	2.521
4.45	.5417	.87533	.87533	2.6478
4.5	.50982	.82248	.82248	2.7523
4.55	.48024	.75195	.75195	2.8292
4.6	.45641	.66518	.66518	2.8736
4.65	.44231	.56487	.56487	2.8815
4.7	.44239	.45509	.45509	2.8503
4.75	.46141	.34141	.34141	2.7794
4.8	.50422	.23083	.23083	2.6707
4.85	.57554	.1317	.1317	2.5289
4.9	.67964	.053395	.053395	2.362
4.95	.82018	.0060244	.0060244	2.1809
5	1	0	0	2

ORIGINAL PAGE IS
OF POOR QUALITY

V3A(INTEGERS(1,35),) (A 35 BY 9 ARRAY)

0	-.0021376	-.079367	-.30641	-.32754	-.078826	.18708	-.32719	-.6673
.05	-.0022661	-.079101	-.30766	-.32793	-.078509	.1885	-.32743	-.66609
.1	-.0024856	-.078891	-.30914	-.32859	-.078266	.18992	-.32795	-.66494
.15	-.0028023	-.078743	-.31087	-.32957	-.07811	.1913	-.3288	-.66393
.2	-.0032195	-.078664	-.31285	-.3309	-.078048	.19262	-.33003	-.66317
.25	-.0037368	-.078654	-.31509	-.33261	-.078088	.19386	-.33168	-.66277
.3	-.0043502	-.078714	-.31757	-.33473	-.07823	.195	-.33376	-.66285
.35	-.0050516	-.078838	-.32028	-.33725	-.078469	.19602	-.33631	-.66351
.4	-.0058287	-.079016	-.3232	-.34017	-.078795	.19694	-.33932	-.66486
.45	-.0066653	-.079236	-.32631	-.34347	-.079192	.19775	-.34279	-.66697
.5	-.0075414	-.079478	-.32958	-.34712	-.079639	.19848	-.34667	-.6699
.55	-.0084337	-.079723	-.33295	-.35107	-.080108	.19915	-.35094	-.67371
.6	-.009316	-.079945	-.33641	-.35525	-.080567	.19981	-.35553	-.6784
.65	-.01016	-.080118	-.33989	-.35961	-.080977	.20051	-.36036	-.68395
.7	-.010937	-.080213	-.34336	-.36405	-.0813	.20131	-.36536	-.6903
.75	-.011616	-.0802	-.34677	-.36848	-.081491	.2023	-.37041	-.69738
.8	-.012171	-.08005	-.3501	-.37283	-.081507	.20356	-.37543	-.70505
.85	-.012573	-.079735	-.3533	-.37699	-.081303	.20519	-.38029	-.71315
.9	-.012801	-.07923	-.35635	-.38089	-.080839	.20725	-.38489	-.72151
.95	-.012837	-.078512	-.35924	-.38444	-.080076	.20988	-.38914	-.72989
1	-.012669	-.077566	-.36197	-.38759	-.078982	.21317	-.39293	-.73807
1.05	-.012294	-.07638	-.36455	-.39029	-.077531	.2172	-.3962	-.7458
1.1	-.011716	-.074951	-.36702	-.39253	-.075708	.22205	-.3989	-.75283
1.15	-.010951	-.073284	-.36941	-.3943	-.073507	.22781	-.40099	-.75891
1.2	-.010025	-.071391	-.37178	-.39566	-.070935	.23452	-.40248	-.76382
1.25	-.0089744	-.069297	-.37423	-.39665	-.068012	.24223	-.4034	-.76736
1.3	-.0078504	-.067034	-.37683	-.3974	-.064772	.25096	-.40384	-.76938
1.35	-.0067143	-.064648	-.3797	-.39804	-.061265	.26068	-.40389	-.76976
1.4	-.0056394	-.062192	-.38296	-.39874	-.05756	.27137	-.40373	-.76848
1.45	-.0047092	-.059732	-.38675	-.39971	-.053737	.28297	-.40353	-.76556
1.5	-.004016	-.057345	-.3912	-.40119	-.049895	.29539	-.40353	-.76111
1.55	-.0036583	-.055115	-.39646	-.40343	-.046147	.3085	-.40399	-.75535
1.6	-.0037379	-.053137	-.40267	-.40672	-.04262	.32214	-.4052	-.74857
1.65	-.004356	-.051509	-.40995	-.41134	-.03945	.33615	-.40748	-.74116
1.7	-.0056095	-.050336	-.41844	-.41759	-.036782	.35032	-.41115	-.73359

C-3

ORIGINAL PAGE IS
OF POOR QUALITY

V3A(INTEGERS(36,70),) (A 35 BY 9 ARRAY)

1.75	-.0075855	-.049723	-.42822	-.42575	-.034764	.36442	-.41656	-.72644
1.8	-.010357	-.049772	-.43938	-.4361	-.033543	.3782	-.42401	-.72034
1.85	-.013977	-.05058	-.45195	-.44887	-.033261	.39141	-.43382	-.71602
1.9	-.018474	-.052233	-.46593	-.46426	-.034045	.40379	-.44625	-.71422
1.95	-.023848	-.054802	-.48126	-.48242	-.036007	.41508	-.46154	-.71574
2	-.030065	-.058337	-.49786	-.50341	-.039232	.42503	-.47984	-.72137
2.05	-.037053	-.062863	-.51556	-.52725	-.043771	.43341	-.50123	-.73189
2.1	-.044703	-.068378	-.53415	-.55383	-.049642	.44003	-.52571	-.74803
2.15	-.052868	-.074844	-.55333	-.58297	-.056815	.44474	-.55318	-.77043
2.2	-.061361	-.082188	-.57277	-.61437	-.065215	.44744	-.58339	-.79964
2.25	-.069961	-.090301	-.59208	-.64762	-.074715	.4481	-.61601	-.83604
2.3	-.078417	-.099032	-.61078	-.68218	-.085139	.44675	-.65054	-.87983
2.35	-.086457	-.1082	-.62839	-.7174	-.096258	.44353	-.68638	-.931
2.4	-.093793	-.11758	-.64438	-.75254	-.1078	.43865	-.72276	-.98929
2.45	-.10014	-.12695	-.65819	-.78674	-.11945	.43241	-.75881	-1.0541
2.5	-.10521	-.13603	-.6693	-.81904	-.13087	.42521	-.79353	-1.1247
2.55	-.10876	-.14457	-.67719	-.84846	-.14172	.41752	-.82586	-1.1998
2.6	-.11057	-.15233	-.68139	-.87395	-.15164	.40987	-.85464	-1.2779
2.65	-.11048	-.15908	-.68155	-.8945	-.16032	.40287	-.87873	-1.3572
2.7	-.10841	-.16466	-.67738	-.90914	-.16749	.3971	-.897	-1.4357
2.75	-.10435	-.16896	-.66878	-.91704	-.17298	.39314	-.90843	-1.5109
2.8	-.098384	-.17199	-.6558	-.91751	-.17668	.39152	-.91215	-1.5804
2.85	-.090696	-.17383	-.63867	-.9101	-.17867	.39261	-.90752	-1.6416
2.9	-.081566	-.17471	-.61783	-.89469	-.17912	.39664	-.89421	-1.6921
2.95	-.071362	-.17499	-.59392	-.87146	-.1784	.4036	-.87225	-1.7296
3	-.060524	-.17513	-.56778	-.84101	-.17705	.41318	-.8421	-1.7521
3.05	-.049548	-.17575	-.54037	-.80433	-.17572	.42476	-.80468	-1.7585
3.1	-.038949	-.17751	-.5128	-.76283	-.17524	.43738	-.76136	-1.748
3.15	-.029231	-.18118	-.4862	-.71827	-.17651	.44972	-.71394	-1.7209
3.2	-.020852	-.1875	-.46167	-.67271	-.18047	.46013	-.66463	-1.6784
3.25	-.014183	-.19719	-.4402	-.62843	-.18801	.46667	-.61587	-1.6224
3.3	-.0094776	-.21083	-.42258	-.58774	-.19996	.46725	-.57027	-1.5563
3.35	-.0068464	-.22887	-.40933	-.55291	-.21695	.45965	-.53039	-1.484
3.4	-.0062408	-.25149	-.40061	-.52593	-.23935	.44174	-.49859	-1.4102
3.45	-.0074496	-.27863	-.39622	-.50839	-.26723	.41157	-.47688	-1.3401

ORIGINAL PAGE IS
OF POOR QUALITY

V3A(INTEGERS(71,101),) (A 31 BY 9 ARRAY)

3.5	-.010108	-.30991	-.39557	-.50135	-.30033	.36753	-.46668	-1.2792
3.55	-.013717	-.34462	-.39768	-.50519	-.33797	.30855	-.46875	-1.233
3.6	-.017682	-.38175	-.40126	-.51952	-.37911	.23415	-.48303	-1.2063
3.65	-.021344	-.41998	-.40475	-.54319	-.42237	.14464	-.5086	-1.2037
3.7	-.024037	-.45774	-.40647	-.57423	-.46602	.041165	-.54369	-1.2282
3.75	-.02513	-.4933	-.40467	-.60998	-.50811	-.074225	-.58567	-1.2822
3.8	-.024084	-.52479	-.39769	-.64707	-.54652	-.19853	-.63114	-1.366
3.85	-.020497	-.55035	-.3641	-.68165	-.57906	-.32782	-.67608	-1.4786
3.9	-.014152	-.5682	-.3628	-.70949	-.60364	-.45727	-.71596	-1.6172
3.95	-.0050542	-.57681	-.33318	-.72618	-.61835	-.58128	-.74596	-1.777
4	.0065397	-.57501	-.29519	-.72738	-.62167	-.69363	-.7612	-1.951
4.05	.020111	-.56217	-.24951	-.70907	-.61264	-.78772	-.75703	-2.1308
4.1	.034889	-.53833	-.19757	-.66786	-.59103	-.85694	-.72937	-2.3056
4.15	.049895	-.50442	-.14166	-.60137	-.55755	-.89516	-.67508	-2.4634
4.2	.064031	-.46232	-.084895	-.50866	-.51397	-.89726	-.59248	-2.591
4.25	.076214	-.41496	-.031176	-.39061	-.46325	-.85987	-.48179	-2.6746
4.3	.085582	-.36629	.01502	-.25043	-.40949	-.78216	-.34567	-2.7012
4.35	.091732	-.32102	.048967	-.093935	-.35773	-.66652	-.18963	-2.6598
4.4	.094997	-.28424	.066211	.070274	-.31349	-.5192	-.022239	-2.543
4.45	.096685	-.26071	.063281	.231	-.28209	-.35052	.14499	-2.3489
4.5	.099199	-.25397	.038555	.37514	-.26764	-.17455	.29829	-2.0824
4.55	.10591	-.26524	-.0068278	.48896	-.2719	-.0079689	.42298	-1.7568
4.6	.12067	-.29242	-.068212	.55999	-.29313	.13187	.5055	-1.3937
4.65	.14677	-.32936	-.13712	.57947	-.32532	.22979	.53599	-1.0214
4.7	.18545	-.36586	-.20171	.5448	-.35819	.27593	.51104	-.67173
4.75	.23385	-.38874	-.24832	.46151	-.37827	.26879	.43582	-.37495
4.8	.28288	-.38411	-.26413	.34401	-.37146	.21694	.32487	-.15417
4.85	.31528	-.34093	-.24077	.21427	-.32681	.13881	.20074	-.019745
4.9	.30445	-.25516	-.17835	.098445	-.24113	.059775	.090166	.034146
4.95	.21428	-.13413	-.089134	.021374	-.12367	.0066075	.018037	.029431
5	0	0	0	0	0	0	0	0

ORIGINAL PAGE IS OF POOR QUALITY