71-24,884

FERG, David Alvin, 1943-
MULTIPLE INPUT-OUTPUT FEEDBACK SYNTHESIS
INCORPORATING CROSS-COUPLING.

The University of Arizona, Ph.D., 1971
Engineering, nuclear

University Microfilms, A XEROX Company, Ann Arbor, Michigan

MULTIPLE INPUT-OUTPUT FEEDBACK SYNTHESIS
INCORPORATING CROSS-COUPLING

by
David Alvin Ferg

A Dissertation Submitted to the Faculty of the
DEPARTMENT OF NUCLEAR ENGINEERING
In Partial Fulfillment of the Requirements
For the Degree of
DOCTOR OF PHILOSOPHY
In the Graduate College
THE UNIVERSITY OF ARIZONA

1 9 7 1

THE UNIVERSITY OF ARIZONA

GRADUATE COLLEGE

I hereby recommend that this dissertation prepared under my
direction by DAVID A. FERG
entitled MULTIPLE INPUT-OUTPUT FEEDBACK SYNTHESIS
INCORPORATING CROSS-COUPLING
be accepted as fulfilling the dissertation requirement of the
degree of DOCTOR OF PHILOSOPHY

Richard L. Bucher
Dissertation Director

April 21, 1971
Date

After inspection of the final copy of the dissertation, the
following members of the Final Examination Committee concur in
its approval and recommend its acceptance:*

R. L. Leach
T. L. Williams
Leonberry
D. L. Schultz
W. Morris Fan

26 April 1971
26 April 1971
26 April 1971
27 April, 1971
27 April, 1971

*This approval and acceptance is contingent on the candidate's adequate performance and defense of this dissertation at the final oral examination. The inclusion of this sheet bound into the library copy of the dissertation is evidence of satisfactory performance at the final examination.

PLEASE NOTE:

**Some pages have indistinct
print. Filmed as received.**

UNIVERSITY MICROFILMS.

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

Signed: David A. Ferg

ACKNOWLEDGMENTS

The author wishes to express his appreciation and indebtedness to Dr. Richard L. Brehm for his direction and assistance during the development of this research. Also, special thanks are extended to Dr. Donald G. Schultz for his advice and comments in Dr. Brehm's absence.

The author is grateful to the United States Atomic Energy Commission for providing financial support for this study under Contract AT-(04-3)-670.

The University of Arizona Computer Center is acknowledged for the generous grant of computer time supplied during the course of the research.

Finally, the author owes a special debt of gratitude to his wife, Maureen, whose unfailing support and patience were a tremendous incentive for the completion of this study.

TABLE OF CONTENTS

	Page
LIST OF ILLUSTRATIONS	vi
LIST OF TABLES	viii
ABSTRACT	ix
1. INTRODUCTION	1
Purpose of Study	2
System Restrictions	4
Content and Organization	5
2. DECOUPLING TECHNIQUES FOR MULTIVARIABLE SYSTEMS	6
System Notation and Definitions	6
System Controllability	9
Integrator Decoupling Results	10
Gilbert's Canonical Decoupled Form	18
Dynamic Subsystem Compensation	31
Procedural Example	36
Summary	55
3. DESIGN OF INTERACTING MULTIVARIABLE CONTROL SYSTEMS USING DECOUPLING TECHNIQUES	57
Variational Decoupling	60
Formulation of Cross-Coupling Control-Law Matrices	74
Computation of D_k Matrices	76
Computational Procedures for Control-Law Realization	80
Summary	105
4. THERMIONIC REACTOR CONTROL SYSTEM	107
Diode Representation	108
Control System Objectives	118
Desired Transfer Matrix Formulation	120
Calculation of Control-Law Matrices	124
Closed-Loop System Time Responses	125
Ramp Change in Reactivity	126
Step change in Diode Load Resistance	131
Ramp in Coolant Temperature	135
Nonlinear Effects	139
Discussion of Results	140

TABLE OF CONTENTS, Continued

	Page
5. CONCLUSIONS	145
Summary of Results	145
Further Research	147
APPENDIX: D_k COMPUTER PROGRAM	151
REFERENCES	185

LIST OF ILLUSTRATIONS

Figure	Page
2-1 Schematic Definition for Desired Closed-Loop Control System for System $S(A,B,C)$	19
2-2 Schematic Definition for Desired Closed-Loop Control System for ID System $\bar{S}(\bar{A},\bar{B},\bar{C})$	20
2-3 Standard Decoupled Form	26
2-4 Canonically Decoupled Form	27
2-5 Feedback Forms for Decoupling	29
2-6 Schematic Definition for Desired Closed-Loop Control System for Gilbert's CD System $\hat{S}(\hat{A},\hat{B},\hat{C})$	30
2-7 Structure of the Decoupled Compensation	33
2-8 Dynamic Series Compensators	34
2-9 Integrator Decoupling Matrices	48
2-10 ID System and CD Transformation Matrices	49
2-11 Canonically Decoupled System Matrices	50
3-1 Canonical Decoupled Structure of $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]$	79
3-2 Original and Partial System Integrator Decoupling Matrices for Example 3.1	88
3-3 Integrator-to-Canonically-Decoupled System Transformation Matrices for Example 3.1	88
3-4 Canonically Decoupled System Matrices for Example 3.1	89
3-5 Canonical System Feedback and Gain Matrices for Example 3.1	91
3-6 Original System Feedback and Gain Matrices for Example 3.1	91

LIST OF ILLUSTRATIONS--Continued

Figure		Page
3-7	D_k Matrices for Example 3.1	92
3-8	$F_k D_k$ Matrices for Example 3.1	93
3-9	Algebraic Equations for Solution of F_T Matrix of Example 3.1	94
3-10	Equations in Row Echelon Form for Solution of F_T Matrix of Example 3.1	94
3-11	$\hat{F}_k$ Matrices for Theorem 3.3	100
3-12	$\hat{F}_k T_k$ Matrices for $k = 1, 2, \dots, m$ of Theorem 3.3	102
4-1	Representation of Regions for Single Diode Approximation .	109
4-2	Linear Responses for +2¢/second Reactivity Ramp of 20-Second Duration	127
4-3	Linear Responses for a -0.007 Ω -cm ² Step Change in Load Resistance	132
4-4	Linear Responses for +1°C/sec Ramp in Coolant Temperature	136
4-5	Nonlinear and Linear Responses of $\delta q_E''$	141
4-6	Nonlinear and Linear Responses of $\delta(IV)$	142

LIST OF TABLES

Table	Page
4-1. Initial Equilibrium Conditions for System	114
4-2. SIMCON Results for Taylor Expansion Coefficients	117
4-3. Design Parameters for Desired Transfer Matrix	122

ABSTRACT

A computational procedure is formulated for the feedback synthesis of a linear multivariable control system with input, output cross-coupling. The feedback synthesis technique which is developed results in a closed-loop interacting transfer matrix containing the desired transfer functions between all permutations of the system outputs and the external control inputs. The technique depends on previously established decoupling techniques for the state-variable feedback design of noninteracting multivariable control systems.

Specifically, the study provides a condensed review of the literature pertaining to decoupling techniques and incorporates the necessary information and formulation required to compute a pair of decoupling state feedback and gain matrices for a multivariable system. A variational decoupling technique is developed which represents a logical mathematical procedure for decoupling a given multivariable system into all possible transfer function relationships between the system outputs and the control inputs.

The variational decoupling technique is used to obtain a restricted set of decoupling matrices. Each pair of decoupling matrices will decouple the system so that each system output is affected by a different control input without repetition over the set of decoupling matrices. The entire set of decoupling matrices are combined into one pair of state-feedback and gain matrices so that

the resultant closed-loop transfer matrix contains the desired control input, system output cross-coupled interaction.

General results for the interacting state-feedback matrix indicate that the matrix does not contain constant gain elements. Consequently, a new form for a state-variable feedback control law is introduced where the simple product of the state-feedback matrix and the state vector is interpreted as a convolution integral over the product.

The applicability of the feedback synthesis technique is indicated by designing a control system for a single diode representation of an in-core thermionic reactor system. Results indicate that a very simple pair of state-feedback and gain matrices will result in a closed-loop control system which can satisfy the following constraints. The diode voltage is restricted to a $\pm 4\%$ variation about an equilibrium value, the thermal power and diode electrical-power dependency is decoupled for sudden load changes, and the thermal shocks on the collector and emitter surfaces are limited to $60^\circ\text{C}/\text{minute}$. The set of input perturbations applied separately to the control system consist of 1) a $2\text{¢}/\text{second}$ reactivity ramp for a time interval of 20 seconds, 2) a 20% step change in the diode load resistance, and 3) a $1^\circ\text{C}/\text{second}$ coolant temperature increase over a time interval of 20 seconds.

CHAPTER 1

INTRODUCTION

A multivariable system is defined in the general sense as a system which has a multiplicity of inputs and outputs and whose impulse response can be described by a set of differential equations. The differential equations which describe a physical dynamic system result from the Newtonian approach to relate mathematically the causal-effect relationships observed in the real world. Using the approach of modern control theory, the basis for the internal mathematical structure of the multivariable system consists of a set of defined state variables. The number of system state variables depends upon the order and number of the differential equations describing the system and represents the minimum amount of information required to know about a system at any time t_0 , such that its future behavior can be determined without reference to the inputs before t_0 (Schultz and Melsa, 1967).

To be physically meaningful, the input and output variables are external to the multivariable system and represent control points of entry to and exit from the multivariable system. The output variables would consist of one or possibly a combination of two or more of the system state variables. Examples of input variables to a multivariable system would be the angular position of the steering wheel and the pivotal displacement of the accelerator on an automobile. Examples

of output variables for the multivariable system represented by an automobile would be its speed and direction.

Control engineering essentially concerns the problem of forcing some physical dynamic system to behave in a manner which meets prescribed performance specifications. Therefore from a control engineer's standpoint, the type of system to be considered is important. The types of mathematical models used to represent physically observable dynamic systems can be classed as being either single-input, single-output representations or multiple-input, multiple output representations. Due to the multiplicity of external forces present and the complexity of presently existing systems, it is not possible to accurately represent most systems using single-input, single-output mathematical models. It is, however, possible to represent small subdivisions of a complex system using many single-input, single-output models. But, designing each single-input, single-output subdivision to respond in a prescribed manner does not consequently result in a desired performance for the entire system because of the inherent interactions between the set of single-input, single-output subsystems. Therefore, in order for the control engineer to achieve the goal of forcing a physically meaningful dynamic system to respond to environmental influences in a desired manner, it is usually imperative to consider multivariable mathematical models.

Purpose of Study

In the last decade, a number of studies dealing with the design of multivariable control systems have appeared in the literature.

Specifically, the studies by Kavanagh (1958), Horowitz (1960), Brockett and Mesarovic (1962), and Morgan (1964) represent early attempts to formulate design procedures in which non-interaction was desired using frequency-domain techniques. The term non-interaction or decoupling is used to describe the situation where each input of the system affects one and only one output without repetition over the set of inputs. Morgan is credited with being the first to use state-variable feedback in the design of a non-interacting multivariable control system. The term state-variable feedback describes the control philosophy where all the system states are combined in the feedback loop and coupled together with the control inputs. However, the culmination of the non-interaction design efforts were achieved by Falb and Wolovich (1967) and by Gilbert (1969). Optimally designed multivariable control systems have been considered by Tyler and Tuteur (1966), Freeman and Abbott (1969), and Davison and Man (1970). Some control studies employing special techniques consist of the use of extended root-locus techniques by Retallack (1970), the inverse Nyquist array by Rosenbrock (1969), minimum sensitivity design via matrix spectral factorization by Bongiorno (1969), and input compensators with output feedback by Brasch and Pearson (1970). Other recent studies include those by Lindgren (1966), Birta and Mufti (1967), Anderson and Luenberger (1967), and Wonham (1967).

The purpose of this study is to present a new method utilizing cross-coupling in the design of closed-loop multivariable control systems. The term cross-coupling denotes the fact that the resultant

control system forces each output of the multivariable system to be affected by all the control inputs in a desired prescribed manner.

From an input-output standpoint, decoupling techniques have the advantage of forcing a multivariable system to function in the simplest possible manner. Also, when the multivariable system is reduced to a set of single-input, single output subsystems, the well-known design techniques presented by Schultz and Melsa (1967) can be used to model each subsystem to achieve a desired response using state-variable feedback and/or dynamic compensators. The objective of this study is to determine a computational procedure for the feedback synthesis of a coupled multivariable control system using the methods of state-variable feedback of non-interacting multivariable control systems.

System Restrictions

Before proceeding with the development, the necessary restrictions required on the mathematical model describing the multivariable system are listed. First, the differential equations constituting the mathematical model must be first-order and linear with time-invariant coefficients. The model must have the same number of control inputs as system outputs, and the number of control inputs must be less than or equal to the number of describing system state variables. The set of equations must also be capable of being decoupled from an input-output point of view and must be controllable.

Content and Organization

The development and illustration of the design technique for an interacting multivariable control system requires first a discussion of the necessary background information, next the mathematical formulation for the interacting feedback synthesis, and finally an application of the resulting technique to a practical example.

Chapter 2 consists of a review of the decoupling techniques resulting in the design of a non-interacting control system. The attention here is focused only on that portion of the existing literature providing the basic information and formulation specifically needed to achieve a decoupled control system using state variable feedback. This framework is supplied by concentrating the review on the work by Falb and Wolovich (1967), Gilbert (1969), and Slivinsky (1969).

Chapter 3 concerns the development of the theory behind the feedback synthesis technique for a multivariable control system with cross-coupling. It consists mainly of two parts; 1) the development of the variational decoupling technique, and 2) the derivation and computation of the feedback and gain matrices resulting in the multivariable control system with desired input-output interaction. The chapter contains four theorems which establish the validity of the design method and several simple examples to help illustrate the procedural steps involved in the design method. This chapter represents the main contributions of the study and constitutes a new approach to the design of multivariable control systems.

CHAPTER 2

DECOUPLING TECHNIQUES FOR MULTIVARIABLE SYSTEMS

The purpose of this chapter is to illustrate the procedural steps for obtaining a decoupled multivariable system. It is not intended that the chapter provide a complete evaluation of all the attempts reported in the literature pertaining to the design of non-interacting multivariable systems. Instead, only that portion of the literature providing essential background material necessary for obtaining a decoupled multivariable system is reviewed.

System Notation and Definitions

In order to apply modern control theory, the linear differential equations which describe the dynamic multivariable system must be represented in vector-matrix notation as follows:

$$\dot{\underline{x}}(t) = \underline{A}\underline{x}(t) + \underline{B}\underline{u}(t) \quad (2-1)$$

$$\underline{y}(t) = \underline{C}\underline{x}(t) \quad (2-2)$$

Here the (·) indicates first-order differentiation with respect to time t and

$\underline{x}(t)$ is an n -dimensional vector, representing the state variables of the system

$\underline{u}(t)$ is an m -dimensional vector, containing the control inputs to the system

$\underline{y}(t)$ is an m -dimensional vector, containing the output variables of the system

A is an $n \times n$ system state matrix

B is an $n \times m$ control input matrix

C is an $m \times n$ output matrix

n is a scalar and denotes the number of state variables representing the system

m is a scalar and denotes the number of control inputs or system outputs

where $m \leq n$ in general. The matrices A , B , and C are constant with no time dependence.

Lower case letters are always used for scalars and vectors, and capital letters always represent matrices. In this section, subscripted lower case letters refer to either a scalar or to an element of a vector, and subscripted capital letters denote submatrices of a matrix. The superscripts T and -1 are to mean the transpose and the inverse of a matrix, respectively. A matrix superscripted with an unparenthesized lower case letter indicates that power of the matrix denoted by the scalar exponent. Also, a superscripted vector with a parenthesized lower case letter indicates the derivative of the vector to the order denoted by the numerical value of the scalar superscript. Capital Greek letters denote vector spaces, and lower case Greek letters indicate elements in the vector space. The symbol I_n is used to represent the $n \times n$ identity matrix, and the symbol $\underline{0}$ is used to indicate either the null vector or the null matrix. Since lower case letters

may denote either vectors or scalars, all vector quantities are underlined in this and subsequent chapters.

The closed-loop control system is obtained by using the control law

$$\underline{u}(t) = \underline{F}\underline{x}(t) + \underline{G}\underline{r}(t) \quad (2-3)$$

where

$\underline{r}(t)$ is an m -dimensional vector, representing the external control inputs to the system

$\underline{G}$ is an $m \times m$ input gain matrix

$\underline{F}$ is an $m \times n$ state feedback matrix

The control law of Eq. (2-3) is defined (Morgan, 1964) as a state-variable feedback control law since the states of the system are "fed back" to the control inputs $\underline{u}(t)$. The closed-loop transfer matrix is obtained in the following manner: substitute Eq. (2-3) into Eq. (2-1) for $\underline{u}(t)$.

$$\dot{\underline{x}}(t) = \underline{A}\underline{x}(t) + \underline{B}\underline{F}\underline{x}(t) + \underline{B}\underline{G}\underline{r}(t) \quad (2-4)$$

Take the Laplace transform of Eq. (2-4), assume zero initial conditions on the state vector $\underline{x}(0)$, and solve for $\underline{x}(s)$,

$$s\mathbf{I}_n \underline{x}(s) = \underline{A}\underline{x}(s) + \underline{B}\underline{F}\underline{x}(s) + \underline{B}\underline{G}\underline{r}(s) \quad (2-5)$$

$$\underline{x}(s) = [s\mathbf{I}_n - \underline{A} - \underline{B}\underline{F}]^{-1} \underline{B}\underline{G}\underline{r}(s) \quad (2-6)$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix. Therefore

$$\underline{y}(s) = \underline{C}\underline{x}(s) = \underline{C}[s\mathbf{I}_n - \underline{A} - \underline{B}\underline{F}]^{-1} \underline{B}\underline{G}\underline{r}(s) \quad (2-7)$$

or, the transfer matrix is:

$$H(s,F,G) = C[sI_n - A - BF]^{-1}BG \quad (2-8)$$

If the transfer matrix $H(s,F,G)$ can be diagonalized by an appropriate choice of F and G , the closed-loop multivariable system is said to be decoupled. The system then possesses non-interaction because each component of the vector input $\underline{r}(t)$ will affect only one component of the vector output $\underline{y}(t)$ due to the diagonal nature of $H(s,F,G)$.

System Controllability

One of the conditions imposed on the type of multivariable systems considered in this study is that the system must be controllable. Therefore, the multivariable system must be first tested to ensure that it is a controllable system (Gilbert, 1963 and Kalman, 1963). If a system is uncontrollable, it has state variables which depend only on initial conditions or disturbance inputs and cannot be affected by the control input vector $\underline{u}(t)$.

The methodology of testing for controllability involves the determination of the rank of the $n \times nm$ matrix $(B, AB, \dots, A^{n-1}B)$, whose columns are the columns of the matrices B, AB , through $A^{n-1}B$. If the rank is n , the system is said to be controllable. The development and proof of this requirement is given in references by Kalman (1963) and Gilbert (1963).

The multivariable system is not required to be observable in this study. If a system is unobservable, it has state variables which are not detectable in the output vector $\underline{y}(t)$ and therefore will have

dynamic transitions in certain state variables which cannot be determined by measurements at the available outputs.

Integrator Decoupling Results

If a single pair of F and G feedback and gain matrices can be calculated which will result in a closed-loop transfer matrix which is nonsingular and diagonal, then it is concluded that the open-loop multi-variable system can be decoupled. Furthermore, if a test can be formulated in terms of the original system matrices, namely A , B , and C , which guarantees that a decoupling pair of matrices does exist, then the test would represent a necessary and sufficient condition for decoupling. The calculation of a pair of F and G matrices from the set of all decoupling matrices necessarily forces a specification of the diagonalized transfer matrix. The specification involves choosing each type of transfer function for the single input, single-output subsystems represented by the diagonal entries in the transfer matrix, $H(s,F,G)$.

Since the transfer functions of the different subsystems are, in general, different from one another in structure, each of the subsystems can be treated independently. The number of states in any subsystem, denoted n_i , will correspond to the number of frequency domain poles in that same subsystem since each state equation is first order (i.e., the subsystem is of order n_i). A desired subsystem response is consequently obtained by appropriate location of each pole in the subsystem through the choice of the elements of the F and G matrices in the control-law of Eq. (2-3) (Schultz and Melsa, 1967).

To determine one pair of F and G matrices from the set of all decoupling matrices, suppose each transfer function for the subsystems of the decoupled multivariable system is specified in the following manner; Let z_i represent the number of zeros in the i^{th} subsystem. Then use z_i of the n_i poles in the i^{th} subsystem to cancel the subsystem zeros and place the remaining $(n_i - z_i)$ poles at the origin. If $(d_i + 1)$ represents the pole-zero excess of the i^{th} subsystem, i.e.,

$$d_i + 1 = n_i - z_i, \quad i = 1, 2, 3, \dots, m \quad (2-9)$$

then the transfer function for the i^{th} subsystem will be

$$y_i(s)/r_i(s) = s^{-(d_i+1)}, \quad i = 1, 2, 3, \dots, m \quad (2-10)$$

where s is the Laplace variable and $r_i(s)$ and $y_i(s)$ are the Laplace transforms of the i^{th} elements in the input and output vectors respectively for the i^{th} subsystem.

Falb and Wolovich (1967) have derived the F and G matrices which will decouple the multivariable system into independent subsystems possessing transfer functions indicated by Eq. (2-10) and the necessary and sufficient condition to determine if the system represented by Eqs. (2-1) and (2-2) can be decoupled into independent subsystems. Although locating all the excess subsystem poles at the origin may not lead to the "most desirable" subsystem characteristics (e.g., weak sensitivity), it does provide an adequate and simple basis for decoupling. Falb and Wolovich (1967) did consider excess pole locations other than at the origin, but this proves to be a less general method of improving subsystem performance than the methods of Gilbert (1969) as will be shown.

The F and G matrices, denoted as F^* and G^* by Falb and Wolovich, are obtained as follows. If the transfer function between the i^{th} output, $y_i(s)$, and the i^{th} external input, $r_i(s)$ is to have no zeros according to Eq. (2-10), there must not be any time derivatives of $r_i(t)$ appearing in the time-domain relationship of $r_i(t)$ and $y_i(t)$. This fact yields the mathematical definition of d_i in terms of the originally defined system matrices, A, B, and C. Substituting the control law of Eq. (2-3) into Eq. (2-1):

$$\dot{\underline{x}}(t) = (A + BF)\underline{x}(t) + BGr(t) \quad (2-11)$$

Then, repeatedly differentiating the output vector p times with respect to time, a derivative relationship between the i^{th} output and the state-variable vector for the i^{th} subsystem is:

$$\dot{y}_i^{(p)} = C_i \dot{x}^{(p)} \quad (2-12)$$

where C_i is the i^{th} row of the output matrix C. After each differentiation of $y_i(t)$, substitute Eq. (2-11) for $\dot{x}(t)$ until the $(d_i+1)^{\text{th}}$ derivative for $y_i(t)$ is obtained. The result is that

$$\begin{aligned} \dot{y}_i^{(d_i+1)} &= C_i (A + BF)^{d_i+1} \underline{x}(t) + C_i (A + BF)^{d_i} BGr \\ &\quad + C_i (A + BF)^{d_i-1} BGr^{(2)} + C_i (A + BF)^{d_i-2} BGr^{(2)} + \dots \\ &\quad + C_i (A + BF) BGr^{(d_i-1)} + C_i BGr^{(d_i)} \end{aligned} \quad (2-13)$$

where $\dot{y}_i^{(p)}$ and $\dot{x}^{(p)}$ denote the p^{th} derivative of the particular variable. To eliminate the derivative terms of $\underline{r}(t)$ in Eq. (2-13) Falb and Wolovich require that d_i be set to the minimum value of a power j such that the product $C_i A^j B \neq 0$, i.e.,

$$d_i = \min[j; C_i A^j B \neq 0, \quad j = 0, 1, \dots, n-1] \quad (2-14)$$

Falb and Wolovich (1967) attach an added condition to the definition of d_i , namely, that if $C_i A^j B$ is zero for all j , then $d_i = n-1$. However, it is apparent that if $C_i A^j B = 0$ for all j , and $d_i = n-1$, then there exists no F and G matrices which can decouple the system. To illustrate, assume that $C_i A^j B = 0$ for all j , and $d_i = n-1$. Since $C_i A^{n-1} B = 0$, Eq. (2-13) becomes:

$$\dot{y}_i^{(n)} = C_i (A + BF)^n x(t)$$

or

$$\dot{y}_i^{(n)} = C_i A^n x(t) \quad (2-15)$$

Therefore, no matter what the choice of F and G , all terms vanish containing F and G due to $C_i A^j B = 0$ for $j = 0, 1, \dots, n-1$. This result was previously noted in a slightly different fashion by Mufti (1969).

Then, using Eq. (2-14), Eq. (2-13) becomes

$$\dot{y}_i^{(d_i+1)} = C_i A^{d_i+1} x(t) + C_i A^{d_i} B F x(t) + C_i A^{d_i} B G r(t) \quad (2-16)$$

Since there are m outputs, $y_i(t)$, m scalar equations with the above form of Eq. (2-16) can be written.

Now define the m -dimensional vector $\underline{y}^*(t)$, the $m \times m$ matrix B^* , and the $m \times n$ matrix A^* as follows:

$$\underline{y}^*(t) = \begin{bmatrix} \dot{y}_1^{(d_1+1)} \\ \dot{y}_2^{(d_2+1)} \\ \vdots \\ \dot{y}_m^{(d_m+1)} \end{bmatrix} \quad B^* = \begin{bmatrix} C_1 A^{d_1} B \\ C_2 A^{d_2} B \\ \vdots \\ C_m A^{d_m} B \end{bmatrix} \quad A^* = \begin{bmatrix} C_1 A^{d_1+1} \\ C_2 A^{d_2+1} \\ \vdots \\ C_m A^{d_m+1} \end{bmatrix} \quad (2-17)$$

where $\underline{y}^*(t)$ is proportional to $\underline{y}(t)$ by way of the Hamiltonian operator for differentiation. Using these definitions for all m scalar equations of the form of Eq. (2-16), the vector-matrix equation for $\underline{y}^*(t)$ is,

$$\underline{y}^*(t) = [A^* + B^*F]\underline{x}(t) + B^*G\underline{r}(t) \quad (2-18)$$

To integrator decouple the multivariable system, it is necessary to define the compensation matrices F and G to be

$$F \equiv F^* = -B^{*-1} A^* \quad (2-19)$$

$$G \equiv G^* = B^{*-1} L \quad (2-20)$$

where L is any $m \times m$ matrix of constant elements which is diagonal and nonsingular. Substitution of Eqs. (2-19) and (2-20) into Eq. (2-18) yields,

$$\underline{y}^*(t) = L\underline{r}(t) \quad (2-21)$$

Since L is diagonal and nonsingular, the system is decoupled, and each $y_i^*(t)$ affects only $r_i(t)$. Now, what is left is to show that the definitions of F^* and G^* will decouple the originally defined system equations (2-1), (2-2), and (2-3).

The i^{th} element $y_i^*(t)$ is the $(d_i+1)^{\text{th}}$ derivative of the i^{th} element $y_i(t)$ of the output vector $\underline{y}(t)$. For zero initial conditions, the Laplace transform of $\underline{y}^*(t)$ in terms of $\underline{y}(t)$ is

$$\underline{y}^*(s) = \begin{bmatrix} s^{d_1+1} y_1(s) \\ s^{d_2+1} y_2(s) \\ \vdots \\ s^{d_m+1} y_m(s) \end{bmatrix} = R(s)\underline{y}(s) \quad (2-22)$$

where

$$R(s) = \begin{bmatrix} s^{d_1+1} & 0 & 0 & \dots & 0 \\ 0 & s^{d_2+1} & 0 & \dots & 0 \\ 0 & 0 & s^{d_3+1} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & s^{d_m+1} \end{bmatrix} \quad (2-23)$$

The output vector $\underline{y}(s)$ written in terms of the input vector, $\underline{r}(s)$ is

$$\underline{y}(s) = R^{-1}(s)\underline{y}^*(s) = R^{-1}(s)L\underline{r}(s) \quad (2-24)$$

But from Eq. (2-7),

$$H(s,F,G) = R^{-1}(s)L = \begin{bmatrix} \ell_1 s^{-d_1-1} & 0 & \dots & 0 \\ 0 & \ell_2 s^{-d_2-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \ell_m s^{-d_m-1} \end{bmatrix} \quad (2-25)$$

Therefore, Eqs. (2-19) and (2-20) represent definitions for a decoupling pair of matrices F and G , since $H(s,F,G)$ is diagonalized. The decoupled system, upon application of F^* and G^* , is said to be integrator decoupled (ID), since to synthesize $y_i(s)/r_i(s)$ on an analog computer, a $(d_i+1)^{th}$ order integrator is required. The i^{th} subsystem will have a transfer function

$$y_i(s)/r_i(s) = h_i(s,F,G) = \ell_i s^{-d_i-1} \quad (2-26)$$

The second main contribution to the decoupling problem by Falb and Wolovich consists of the test to determine if the originally given multivariable system can be decoupled. An examination of Eqs. (2-19) and (2-20) reveals that to use F^* and G^* as a decoupling pair, the nonsingularity of B^* is explicitly assumed. The nonsingularity of B^* represents the necessary and sufficient condition for determining whether a given multivariable system can be decoupled with state-variable feedback into subsystems possessing noninteraction. The sufficiency condition of B^* nonsingular has been illustrated by showing that the F and G matrices given by Eqs. (2-19) and (2-20) do decouple the system. The necessary condition can be shown by the following arguments. It was previously noted that if there exist matrices F and G which decouple the multivariable system, then for each i ,

$$C_i A^{d_i} B \neq 0 \quad i = 1, 2, \dots, m \quad (2-27)$$

for some d_i . Equation (2-16) indicates that if each element in the input vector $\underline{r}(t)$ is to affect one and only one element in the output vector $\underline{y}(t)$, then

$$C_i A^{d_i} B G = \lambda_{d_i} e_i \quad (2-28)$$

where e_i denotes the i^{th} row of the $m \times m$ identity matrix, I_m . Now λ_{d_i} cannot be zero since λ_{d_i} equal to zero implies that because G is nonsingular and diagonal, $C_i A^{d_i} B = 0$, which contradicts the fact that F and G decoupling matrices do exist. Repeating the argument for each row of the output vector and combining all m resultant equations of the form of Eq. (2-28), then

$$\begin{bmatrix} C_1 A^{d_1} B \\ C_2 A^{d_2} B \\ \vdots \\ C_m A^{d_m} B \end{bmatrix} G = L = B^* G \quad (2-29)$$

So, since G and L are diagonal and nonsingular

$$B^{*-1} = GL^{-1} \quad (2-30)$$

and it is concluded that B^* nonsingular is a necessary condition for the system to be decoupled. More complete and rigorous proofs of the necessary and sufficient condition on B^* for decoupling are given by Falb and Wolovich (1967) and Mufti (1969).

The use of the pair of matrices, F^* and G^* results in a decoupled multivariable system which has several disadvantages. Namely, each subsystem has one or more poles only at the origin and no zeros. These shortcomings cause highly impractical features from a design standpoint since zeros are required to obtain a zero velocity-error coefficient for step inputs and zero position-error coefficients for ramp inputs (Kuo, 1962) and only closed-loop time-responses of the form

$$y_i(t) = a_i t^{d_i} r_i(t) \quad (2-31)$$

exist. Also, since the cancellation of all subsystem zeros with some of the subsystem poles is an inherent consequence of using F^* and G^* , those poles used in the cancellation cannot be arbitrarily positioned using state-variable feedback. This fact, as shown in Schultz and Melsa

(1967), always results in a suboptimally designed closed-loop system. Also, if the open-loop system is observable, the closed-loop system will be unobservable.

Gilbert's Canonical Decoupled Form

With the ability to determine whether a multivariable system can be decoupled by state-variable feedback and with the restricted class of decoupling matrices which integrator decouple, Gilbert (1969) proceeded to formulate a methodology extending Falb and Wolovich's work in order to avoid the disadvantages just mentioned. Namely, he provided answers to the questions: 1) if decoupling is possible, what freedom of choice exists for the F and G control-law matrices, 2) what kinds of decoupled closed-loop responses are attainable, and 3) once an attainable closed-loop response is specified, how are the F and G matrices computed?

Gilbert first integrator-decoupled the system using the F^* and G^* matrices defined by Eqs. (2-19) and (2-20). Then, treating this integrator decoupled (ID) system as an open-loop system, he proceeded to find appropriate feedback and gain matrices to produce a desired response.

The basis for this approach arises as follows. Let $S(A,B,C)$ represent the original open-loop multivariable system defined by Eqs. (2-1) and (2-2) and define F and G to be matrices for the state-variable control law of Eq. (2-3) used to compensate $S(A,B,C)$ (see Fig. 2-1). Next, let $\bar{S}(\bar{A},\bar{B},\bar{C})$ represent the ID system found by compensating $S(A,B,C)$ with F^* and G^* and which is subsequently treated as an open-loop system (see Fig. 2-2). Define $\bar{F}$ and $\bar{G}$ to be the matrices used to

Fig. 2-1. Schematic Definition for Desired Closed-Loop Control System for System $S(A,B,C)$

Fig. 2-2. Schematic Definition for Desired Closed-Loop Control System for ID System $\bar{S}(\bar{A}, \bar{B}, \bar{C})$

compensate $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ to obtain a desired closed-loop response as shown in Fig. (2-2). The multivariable systems $S(A, B, C)$ and $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ are control-law-equivalent if a one-to-one correspondence between F and $\bar{F}$, G and $\bar{G}$ can be found such that for this correspondence $H(s, F, G)$ equals $\bar{H}(s, \bar{F}, \bar{G})$, where $H(s, F, G)$ and $\bar{H}(s, \bar{F}, \bar{G})$ are the transmission matrices [Eq. (2-7)] relating the output vector $y(t)$ to the input vector $r(t)$ for $S(A, B, C)$ and $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ respectively.

The state equations for $S(A, B, C)$ are the familiar ones of Eqs. (2-1) and (2-2). The state equations for $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ are

$$\dot{\bar{x}}(t) = \bar{A}\bar{x}(t) + \bar{B}\bar{u}(t) \quad (2-32)$$

$$y(t) = \bar{C}\bar{x}(t) \quad (2-33)$$

where

$$\bar{A} = A + BF^* \quad (2-34)$$

$$\bar{B} = BG^* \quad (2-35)$$

$$\bar{C} = C \quad (2-36)$$

and the control-vector $\bar{u}(t)$ represents the point of application for the next feedback compensation. But F^* and G^* are defined by Eqs. (2-19) and (2-20) so that upon substitution,

$$\dot{\bar{x}}(t) = (A - BB^{*-1}A^*)\bar{x}(t) + BB^{*-1}\bar{u}(t) \quad (2-37)$$

Using the previously defined $\bar{F}$ and $\bar{G}$ matrices, the control law input for $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ is therefore,

$$\bar{u}(t) = \bar{F}\bar{x}(t) + \bar{G}r(t) \quad (2-38)$$

and Eq. (2-37) becomes

$$\dot{\underline{x}}(t) = (A - BB^{*-1}A^* + BB^{*-1}\bar{F})\underline{x}(t) + BB^{*-1}\bar{G}r(t) \quad (2-39)$$

The corresponding expression for $S(A,B,C)$ using Eq. (2-3) is

$$\dot{\underline{x}}(t) = (A + BF)\underline{x}(t) + BGr(t) \quad (2-40)$$

The systems $S(A,B,C)$ and $\bar{S}(\bar{A},\bar{B},\bar{C})$ will have identical responses if

Eqs. (2-39) and (2-40) are identical. This requires that

$$\begin{aligned} F &= B^{*-1}(\bar{F} - A^*) \\ F &= B^{*-1}\bar{F} + F^* \end{aligned} \quad (2-41)$$

$$G = B^{*-1}\bar{G} \quad (2-42)$$

By this development it is established that one can work with either the original system or the ID system and still retain the same design freedom. Gilbert chose to work with the ID system because it is decoupled.

After integrator decoupling the system but before attempting to determine an appropriate $\bar{F}$ and $\bar{G}$ matrix pair, Gilbert found a canonical transformation which would place the ID system in a unique form which allows identification of each of the m subsystems. The canonical transformation is performed on the ID system rather than the original system because first, the ID system is decoupled and therefore simpler in structure, and second, the canonical forms for coupled multivariable systems [e.g., Luenberger (1967) and Asseo (1968)] do not answer questions 1), 2), and 3) adequately and consistently. The subsystem identification consists of determining the particular control input and subsystem output, the number of subsystem states or subsystem frequency-domain poles, and the number and location of the subsystem zeros.

Consequently, the identification of all the m subsystems establishes the restrictions of state-variable feedback, since state-variable feedback cannot add new zeros or control the location of existing ones, and it cannot increase the order of any subsystem.

The canonical form is developed along the following lines. The ID system and the canonical decoupled (CD) system are related by a linear change of variables using the transformation matrix T , where

$$\hat{\underline{x}}(t) \equiv T\underline{x}(t) \quad (2-43)$$

The $\hat{\underline{x}}(t)$ represents the state-variable vector for the CD system, and $\underline{x}(t)$ is the state variable vector for both the original and the ID system.

The theoretical considerations and required proofs connected with Gilbert's formulation of the transformation matrix T are not examined in this study. Instead, only the basic computational procedures necessary to calculate the matrix T are presented because they are pertinent to the application of multivariable decoupling techniques.

The symbol T denotes the n -dimensional space of n -element row matrices. Also, τ_i represents any one of the n -element row matrices in the space T , where in general

$$\tau_i = [\tau_{1_i}, \tau_{2_i}, \tau_{3_i}, \dots, \tau_{n_i}] \quad (2-44)$$

Now, each τ_i matrix for $i = 1, 2, \dots, m$ is computed by solving the $(m-1) \times n$ equations,

$$\tau_i \bar{A}^j \bar{B}_k = 0 \quad (2-45)$$

for $k = 1, 2, \dots, m$, $k \neq i$ and $j = 0, 1, 2, \dots, n-1$. The $\bar{A}$ and $\bar{B}$ matrices

are defined by Eqs. (2-34) and (2-35), and $\bar{B}_k$ denotes the k^{th} column of the matrix $\bar{B}$.

The reason for computing the τ_i row matrices is that two scalars, p_i and ℓ_i , are needed in the definition of the CD representation, and they are defined as

$$p_i = \text{dimension of } \tau_i, \quad i = 1, 2, \dots, m \quad (2-46)$$

where

$$p = \sum_{i=1}^m p_i \quad (2-47)$$

and

$$p_{m+1} = n - p \quad (2-48)$$

$$\ell_i = p_i - d_i - 1 \quad (2-49)$$

with d_i defined by Eq. (2-14). The scalar p_i is the number of poles in the i^{th} subsystem which can be controlled by state-variable feedback, and ℓ_i is the number of fixed zeros of the i^{th} subsystem.

The transformation matrix T is written as

$$T = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{m+1} \end{bmatrix} \quad (2-50)$$

where the rows of the $(p_i \times n)$ matrix T_i , $i = 1, 2, \dots, m$, are a basis for τ_i . The first $d_i + 1$ rows of each of these T_i are chosen to be $C_i, C_i \bar{A}, \dots, C_i \bar{A}^{d_i}$ and the last $p_i - d_i - 1$ rows are any other linearly independent row vectors which make the set of rows of T_i a basis for τ_i .

For T_{m+1} , the rows are chosen so that the collection of rows of T form a basis for τ . A computer program has been written by Gilbert and Pivnichny (1969) which incorporates the computation of the transformation matrix, T . Consequently, the tedious and difficult procedure of calculating the T matrix by longhand is bypassed. However, a simple example presented at the end of this chapter will illustrate the basic calculations involved in constructing the ID system to CD system transformation matrix, T .

Substitution of Eq. (2-43) into Eqs. (2-32), (2-33), and (2-38) yields

$$\dot{\tilde{x}}(t) = TAT^{-1}\tilde{x}(t) + TB\tilde{u}(t) \quad (2-51)$$

$$\tilde{y}(t) = CT^{-1}\tilde{x}(t) \quad (2-52)$$

$$\tilde{u}(t) = FT^{-1}\tilde{x}(t) + G\tilde{r}(t) \quad (2-53)$$

Define:

$$\hat{A} = TAT^{-1} \quad (2-54)$$

$$\hat{B} = TB$$

$$\hat{C} = CT^{-1}$$

and

$$\hat{F} = FT^{-1} \quad (2-55)$$

$$\hat{G} = G$$

The canonical transformation is completed if the $\hat{A}$, $\hat{B}$, $\hat{C}$, $\hat{F}$, and $\hat{G}$ matrices can be partitioned into specific forms. The $\hat{A}$, $\hat{B}$, and $\hat{C}$ matrices must have the partitioned standard decoupled form as shown in Fig. 2-3, and the subsystems $\hat{A}_{ii}$, $\hat{B}_{ii}$, and $\hat{C}_{ii}$ must exhibit the canonically decoupled form illustrated in Fig. 2-4. The diagonal nature

$$\hat{A} = \begin{bmatrix} \hat{A}_{11} & 0 & 0 & \dots & 0 & 0 \\ 0 & \hat{A}_{22} & 0 & \dots & 0 & 0 \\ 0 & 0 & \hat{A}_{33} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \hat{A}_{mm} & 0 \\ \hat{A}_1^c & \hat{A}_2^c & \hat{A}_3^c & \dots & \hat{A}_m^c & \hat{A}_{m+1}^c \end{bmatrix}$$

where $\hat{A}_{ii}$ is a $p_i \times p_i$ matrix
 $\hat{A}_i^c$ is a $p_{m+1} \times p_i$ matrix

$$\hat{B} = \begin{bmatrix} \hat{B}_{11} & 0 & 0 & \dots & 0 \\ 0 & \hat{B}_{22} & 0 & \dots & 0 \\ 0 & 0 & \hat{B}_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \hat{B}_{mm} \\ \hat{B}_1^c & \hat{B}_2^c & \hat{B}_3^c & \dots & \hat{B}_m^c \end{bmatrix}$$

where $\hat{B}_{ii}$ is a $p_i \times 1$ submatrix
and $\hat{B}_i^c$ is a $p_{m+1} \times 1$ submatrix

$$\hat{C} = \begin{bmatrix} \hat{C}_{11} & 0 & 0 & \dots & 0 & 0 \\ 0 & \hat{C}_{22} & 0 & \dots & 0 & 0 \\ 0 & 0 & \hat{C}_{33} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \hat{C}_{mm} & 0 \end{bmatrix}$$

where $\hat{C}_{ii}$ is a $1 \times p_i$ submatrix
 $p_i \geq d_i + 1, \quad i=1,2,\dots,m$
 $p_i = \ell_i + d_i + 1$

Fig. 2-3. Standard Decoupled Form

$$\hat{A}_{ii} = \begin{bmatrix} \begin{bmatrix} 0 & I_{d_i} \\ 0 & 0 \end{bmatrix} & 0 \\ V_i & W_i \end{bmatrix}$$

where I_{d_i} is the $d_i \times d_i$ identity submatrix

V_i is an $\ell_i \times (d_i + 1)$ submatrix

W_i is an $\ell_i \times \ell_i$ submatrix

$$\hat{B}_{ii} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ b_i \end{bmatrix} \quad \text{where } b_i \text{ is an } \ell_i \times 1 \text{ submatrix}$$

$$\hat{C}_{ii} = [1 \quad 0 \quad 0 \quad \dots \quad 0]$$

Fig. 2-4. Canonically Decoupled Form

of the upper portions of the $\hat{A}$ and $\hat{B}$ matrices and the left portion of the $\hat{C}$ matrix permit the identification of m decoupled subsystems. Gilbert (1969) proved that upon using the correctly constructed transformation matrix T , the matrices A , B , and C have the structure required for the canonically decoupled system representation.

Gilbert has also shown that F and G will preserve noninteraction if and only if the matrices $\hat{F}$ and $\hat{G}$ related by Eqs. (2-56) and (2-57) have the partitioned forms shown in Fig. 2-5.

To summarize, after the original system has been integrator decoupled (ID), the resulting ID system is considered to be an open loop system with control input vector $\bar{u}(t)$. This ID system is to be further compensated with $\bar{F}$ and $\bar{G}$ to obtain the desired diagonal transmission matrix. Before this control law can be used as compensation, a canonical transformation is performed to identify the m subsystems. Then, each subsystem is designed to have a particular response using well-known single-input, single-output state-variable feedback design techniques (Schultz and Melsa, 1967). The final compensated CD system is illustrated in Fig. 2-6. A single pair of feedback and gain matrices for the open-loop system are computed using the equations below.

$$F = F^* + B^{*-1} \hat{F} T \quad (2-56)$$

$$G = B^{*-1} \hat{G} \quad (2-57)$$

It is pointed out that the feedback matrix F consists of one part which destroys the static coupling inherent in the open loop system and another part which models the subsystems for a particular closed-loop

$$\hat{F} = \begin{bmatrix} \hat{F}_{11} & 0 & 0 & \dots & 0 & 0 \\ 0 & F_{22} & 0 & \dots & 0 & 0 \\ 0 & 0 & F_{33} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & F_{mm} & 0 \end{bmatrix}$$

where $\hat{F}_{ii}$ is a $1 \times p_i$ submatrix

$$\hat{G} = \begin{bmatrix} \hat{G}_{11} & 0 & 0 & \dots & 0 \\ 0 & \hat{G}_{22} & 0 & \dots & 0 \\ 0 & 0 & \hat{G}_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \hat{G}_{mm} \end{bmatrix}$$

where $\hat{G}_{ii}$ is a 1×1 submatrix

and $\hat{G}_{ii} \neq 0, \quad i = 1, 2, \dots, m$

Fig. 2-5. Feedback Forms for Decoupling

Fig. 2-6. Schematic Definition for Desired Closed-Loop Control System for Gilbert's CD System $\hat{S}(\hat{A}, \hat{B}, \hat{C})$

response. Application of the F and G matrices to the original open-loop system results in a closed-loop system which has a diagonal transfer matrix containing the desired subsystem transfer functions.

With this contribution, Gilbert has reduced the complex design problem for multivariable plants to a systematic methodology. A further extension of this methodology has been provided by Slivinsky (1969) who considered the possibility and consequence of adding dynamic compensation to the subsystems of the ID system. His contribution to this important aspect is discussed next.

Dynamic Subsystem Compensation

In general, the number of poles and zeros in any one subsystem may be inadequate for obtaining a desired subsystem response. For example, suppose that a particular subsystem of a canonically decoupled multivariable system possesses two poles and a zero. The control engineer, however, desires that the particular subsystem exhibit a closed-loop response which requires three poles. Also, the zero is to be relocated in order to achieve a zero position error for applied ramp inputs to the closed-loop subsystem. Since state-variable feedback by itself does not possess the ability to increase the order of a system (adding a subsystem pole) or to control the position of an improperly located zero, the subsystem control objective cannot be satisfied.

The means for removing these types of inadequacies is to insert lead-lag compensators in the control-input channels of the subsystems. Slivinsky (1969) has proven that when dynamics in the form of lead-lag compensators are added to the decoupled subsystems, they do not cause a

loss of decoupling. State variable feedback then permits the arbitrary positioning of all the poles in the augmented subsystem. The development presented by Slivinsky (1969) of the procedures required to add series compensation to a multivariable system constitutes the remainder of the section.

After the original system, $S(A,B,C)$, is integrator decoupled, the system represented by $\bar{S}(\bar{A},\bar{B},\bar{C})$ is transformed to the canonically-decoupled form represented by $\hat{S}(\hat{A},\hat{B},\hat{C})$. In the canonically decoupled form, the subsystems are identified and, depending upon the number of zeros and poles in each subsystem and the desired subsystem responses, the type and amount of compensation are determined.

The equations describing the decoupled compensation are,

$$\ddot{\tilde{x}}(t) = \ddot{\tilde{A}}\tilde{x}(t) + \ddot{\tilde{B}}\tilde{u}(t) \quad (2-58)$$

$$\bar{u}(t) = \check{\tilde{C}}\tilde{x}(t) + \check{\tilde{E}}\tilde{u}(t) \quad (2-59)$$

The $\bar{u}(t)$ vector represents the control entry point to the integrator decoupled system, $\bar{S}(\bar{A},\bar{B},\bar{C})$ and the $\tilde{u}(t)$ vector consequently represents the control entry point to the augmented system which consists of the ID system plus the added decoupled compensation. The compensation state-variable vector, $\tilde{x}(t)$, represents the number of added states of compensation to the ID system and has dimension n_c . The term decoupled compensation is used because the $\check{\tilde{E}}$ matrix is diagonal, and coupling between input channels is purposely omitted. The structure for the decoupled compensation is illustrated in Fig. 2-7. An example of a general n_c^{th} -order series compensator is shown in block diagram form in Fig. 2-8a.

$$\check{A} = \begin{bmatrix} \check{A}_{11} & 0 & \dots\dots & 0 \\ 0 & \check{A}_{22} & \dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots & \check{A}_{mm} \end{bmatrix}$$

where $\check{A}_{ii}$ is an $n_{c_i} \times n_{c_i}$ submatrix, and

$\check{A}$ is an $n_c \times n_c$ matrix

$$\check{B} = \begin{bmatrix} \check{B}_{11} & 0 & \dots\dots & 0 \\ 0 & \check{B}_{22} & \dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots & \check{B}_{mm} \end{bmatrix}$$

where $\check{B}_{ii}$ is an $n_{c_i} \times m$ submatrix, and

$\check{B}$ is an $n_c \times m$ matrix

$$\check{C} = \begin{bmatrix} \check{C}_{11} & 0 & \dots\dots & 0 \\ 0 & \check{C}_{22} & \dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots & \check{C}_{mm} \end{bmatrix}$$

where $\check{C}_{ii}$ is a $1 \times n_{c_i}$ submatrix, and

$\check{C}$ is an $m \times n_c$ matrix

$$\check{E} = \begin{bmatrix} \check{E}_{11} & 0 & \dots\dots & 0 \\ 0 & \check{E}_{22} & \dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots & \check{E}_{mm} \end{bmatrix}$$

where $\check{E}_{ii}$ is a 1×1 submatrix, and

$\check{E}$ is an $m \times m$ matrix

Fig. 2-7. Structure of the Decoupled Compensation

a) Block Diagram Representation of Dynamic Series Compensation

b) Compensation Added to Procedural Example

Fig. 2-8. Dynamic Series Compensators

The compensation is added to the control input channels of the ID system $\bar{S}(\bar{A}, \bar{B}, \bar{C})$ and the resultant augmented system represented by $\tilde{S}(\tilde{A}, \tilde{B}, \tilde{C})$ is obtained. The state-variable vector for the augmented system is defined to be

$$\tilde{\underline{x}}(t) = \begin{bmatrix} \underline{x}(t) \\ \underline{\tilde{x}}(t) \end{bmatrix} \quad (2-60)$$

where $\underline{x}(t)$ and $\underline{\tilde{x}}(t)$ are the state vectors for the ID system and for the decoupled compensation, respectively. The dimension of $\tilde{\underline{x}}(t)$ is $(n+n_c)$. From the definition for $\tilde{\underline{x}}(t)$, the state equations for the augmented multivariable system $\tilde{S}(\tilde{A}, \tilde{B}, \tilde{C})$ are

$$\dot{\tilde{\underline{x}}}(t) = \tilde{A}\tilde{\underline{x}}(t) + \tilde{B}\underline{u}(t) \quad (2-61)$$

$$\underline{y}(t) = \tilde{C}\tilde{\underline{x}}(t) \quad (2-62)$$

where in terms of the ID system matrices and the decoupled compensation matrices,

$$\tilde{A} = \begin{bmatrix} \bar{A} & \bar{B}\bar{C} \\ 0 & \tilde{A} \end{bmatrix} \quad (2-63)$$

$$\tilde{B} = \begin{bmatrix} \bar{B}\tilde{E} \\ \tilde{B} \end{bmatrix} \quad (2-64)$$

$$\tilde{C} = [\bar{C} \quad 0] \quad (2-65)$$

The design procedure is then repeated by considering the augmented multivariable system $\tilde{S}(\tilde{A}, \tilde{B}, \tilde{C})$ the same as the original uncompensated multivariable system $S(A, B, C)$. If the control law used to obtain a closed-loop desired transfer matrix for the augmented multivariable system is denoted

$$\tilde{u}(t) = \tilde{F}\tilde{x}(t) + \tilde{G}r(t) \quad (2-66)$$

then, Eqs. (2-56) and (2-57) become

$$\tilde{F} = F^* + B^{*-1} \hat{F}T \quad (2-67)$$

and

$$\tilde{G} = B^{*-1} \hat{G} \quad (2-68)$$

In other words, $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, $\tilde{F}$, and $\tilde{G}$ replace A , B , C , F , and G respectively in the design equations for the situation where the originally given multivariable system must be compensated to obtain a particular desired closed-loop transfer matrix.

A simple example will illustrate the design procedures presented in the previous three sections of this chapter.

Procedural Example

The example of a reactor system with two temperature reactivity feedback utilizing the simple model employed in the stability studies by Brehm, Hetrick, et al., (1968) will illustrate the procedural steps and basic calculations involved in decoupling a multivariable system. The state-variables which are assumed to be externally controllable are

the neutron power and the diode collector temperature. The state variables representing measurable output variables of the system are again the neutron power and the diode collector temperature. The following first-order differential equations then represent the mathematical model of the multivariable system.

$$\dot{x}_1(t) = 250 x_2(t) - 500 x_3(t) + u_1(t) \quad (2-69)$$

$$\dot{x}_2(t) = 0.26 x_1(t) - 0.031 x_2(t)$$

$$\dot{x}_3(t) = 0.55 x_2(t) - 8.65 x_3(t) + u_2(t)$$

$$y_1(t) = x_1(t)$$

$$y_2(t) = x_3(t)$$

where

$x_1(t)$ = state-variable of neutron power density

$x_2(t)$ = state-variable of emitter-surface temperature

$x_3(t)$ = state-variable of collector-surface temperature

The state-variable, control, and output vectors are defined as

$$\underline{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} \quad \underline{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \quad \underline{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} \quad (2-70)$$

and the associated system matrices are:

$$A = \begin{bmatrix} 0 & 250 & -500 \\ 0.26 & -0.031 & 0 \\ 0 & 0.55 & -8.65 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-71)$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-71)$$

The decoupling procedure for any multivariable system would consist of the following steps.

a) Determine the scalars d_i defined by Eq. (2-14).

For this example, $m = 2$, and $n = 3$. Therefore, only d_1 and d_2 are needed because there are only two rows to the output matrix C , namely

$$C_1 = [1 \quad 0 \quad 0]$$

$$C_2 = [0 \quad 0 \quad 1]$$

Consider first d_1 and $j = 0$, for $i = 1$.

$$CB = [1 \quad 0 \quad 0] \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = [1 \quad 0] \neq \text{null matrix}$$

so, $d_1 = 0$.

Consider d_2 and $j = 0$, for $i = 2$.

$$C_2 B = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \end{bmatrix} \neq \text{null matrix}$$

so, $d_2 = 0$.

b) Determine if the system can be decoupled by calculating B^* as defined by Eq. (2-17).

Since d_1 and d_2 are both zero,

$$B^* = \begin{bmatrix} C_1 B \\ C_2 B \end{bmatrix} = CB$$

or

$$B^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-72)$$

Since B^* equals the (2×2) identity matrix, the inverse of B^* , B^{*-1} , does exist and is

$$B^{*-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-73)$$

and therefore, the multivariable system can be decoupled using state-variable feedback.

c) Determine the A^* matrix as defined by Eq. (2-17).

Since d_1 and d_2 are both zero,

$$A^* = \begin{bmatrix} C_1 A \\ C_2 A \end{bmatrix} = CA$$

or,

$$A^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 250 & -500 \\ 0.26 & -0.031 & 0 \\ 0 & 0.55 & -8.65 \end{bmatrix} = \begin{bmatrix} 0 & 250 & -500 \\ 0 & 0.55 & -8.65 \end{bmatrix} \quad (2-74)$$

d) Determine F^* and G^* , the compensation matrices which put the system in integrator decoupled (ID) form, defined by Eqs. (2-19) and (2-20).

Since B^{*-1} equals the (2x2) identity matrix,

$$F^* = -B^{*-1} A^* = -A^* = \begin{bmatrix} 0 & -250 & 500 \\ 0 & -0.55 & 8.65 \end{bmatrix} \quad (2-75)$$

and

$$G^* = B^{*-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-76)$$

e) Determine the $\bar{A}$, $\bar{B}$, and $\bar{C}$ matrices defined by Eq. (2-34), (2-35), and (2-36).

Now, if $\bar{A} = A + BF^*$ then

$$\bar{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0.26 & -0.031 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2-77)$$

and if $\bar{B} = BG^*$

$$\bar{B} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-78)$$

and $\bar{C} = C$ where C is given by Eq. (2-71).

f) Determine the matrix T , used to change variables and to determine the canonically decoupled representation of the ID system, defined by Eq. (2-43).

From Eq. (2-45), the vector spaces τ_1 and τ_2 must be calculated since $m = 2$. Consequently, the two equations,

$$\tau_1 \bar{A}^j \bar{B}_2 = 0, \quad j = 0, 1, 2 \quad (2-79)$$

$$\tau_2 \bar{A}^j \bar{B}_1 = 0, \quad j = 0, 1, 2$$

must be solved for the values of j indicated to obtain the row matrices τ_1 and τ_2 where, since $n = 3$,

$$\tau_i = [\tau_{i1} \ \tau_{i2} \ \tau_{i3}] , \quad i = 1, 2$$

For $j = 0$,

$$\tau_1 \bar{B}_2 = \tau_{13} = 0$$

$$\tau_2 \bar{B}_1 = \tau_{2_1} = 0$$

For $j = 1$,

$$\tau_1 \bar{A} \bar{B}_2 = 0$$

$$\tau_2 \bar{A} \bar{B}_1 = 0.26 \tau_{2_2} = 0, \text{ so } \tau_{2_2} = 0$$

For $j = 2$,

$$\tau_1 A^2 B_2 = 0$$

$$\tau_2 \bar{A}^2 \bar{B}_1 = 0$$

so,

$$\tau_1 = [\tau_{1_1}, \tau_{1_2}, 0]$$

$$\tau_2 = [0, 0, \tau_{2_3}]$$

By definition, the dimension of τ_1 is p_1 and therefore,

$$p_1 = \dim \tau_1 = 2$$

Also, the dimension of τ_2 is p_2 and so,

$$p_2 = \dim \tau_2 = 1$$

The dimensions of τ_1 and τ_2 add to 3, the number of system states, so $p_{m+1} = 0$, and only T_1 and T_2 are needed to construct T . Since d_1 and d_2 are both zero, the first row of T_1 is C_1 and the first row of T_2 is C_2 . The second row of T_1 is any row matrix which is independent of C_1 and belongs to the subspace of τ_1 . Choose the row matrix $[0 \ 1 \ 0]$

because if $\tau_{1_1} = 0$ and $\tau_{1_2} = 1$, then

$$\tau_1 = [0 \ 1 \ 0]$$

and so the row matrix $[0 \quad 1 \quad 0]$ belongs to the subspace of τ_1 .

Therefore,

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I_3 \quad (2-80)$$

$$T^{-1} = \text{inverse } T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-81)$$

g) Determine the canonical system matrices $\hat{A}$, $\hat{B}$, and $\hat{C}$ defined by Eq. (2-54).

$$\hat{A} = T\bar{A}T^{-1} = \bar{A} \text{ since } T = T^{-1} = I_3$$

$$\hat{B} = T\bar{B} = \bar{B} \text{ since } T = I_3$$

$$\hat{C} = \bar{C}T^{-1} = \bar{C} \text{ since } T = T^{-1} = I_3$$

So,

$$\hat{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0.26 & -0.031 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \hat{B} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-82)$$

$$\hat{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

h) Determine the integer, ℓ_i , for each subsystem in order to partition $\hat{A}$, $\hat{B}$, and $\hat{C}$.

From step f), p_1 is 2, and p_2 is 1, so using Eq. (2-49)

$$l_1 = p_1 - d_1 - 1 = 2 - 1 = 1 \quad d_1 = 0$$

$$l_2 = p_2 - d_2 - 1 = 1 - 1 = 0 \quad d_2 = 0$$

i) Partition $\hat{A}$, $\hat{B}$, and $\hat{C}$, identify the subsystems, and determine the number of zeros and poles in each subsystem.

$$\hat{A} = \begin{bmatrix} 0 & 0 & | & 0 \\ 0.26 & -0.031 & | & 0 \\ \hline 0 & 0 & | & 0 \end{bmatrix} \quad \hat{B} = \begin{bmatrix} 1 & | & 0 \\ 0 & | & 0 \\ \hline 0 & | & 1 \end{bmatrix} \quad (2-83)$$

$$\hat{C} = \begin{bmatrix} 1 & 0 & | & 0 \\ \hline 0 & 0 & | & 1 \end{bmatrix}$$

So,

$$\hat{A}_{11} = \begin{bmatrix} 0 & 0 \\ 0.26 & -0.031 \end{bmatrix} \quad \hat{A}_{22} = [0] \quad (2-84)$$

$$\hat{B}_{11} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \hat{B}_{22} = [1]$$

$$\hat{C}_{11} = [1 \ 0] \quad \hat{C}_{22} = [1]$$

So, subsystem 1 has one zero and two poles, and subsystem 2 has only one pole and no zeros.

j) Choose the desired closed-loop subsystem transfer functions and add the necessary compensation if the number of subsystem poles and zeros is inadequate to achieve the desired response.

For this example, suppose the desired transfer function for the the first subsystem, denoted $h_{d_1}(s)$ is

$$h_{d_1}(s) = \frac{2}{s^2 + 2s + 2} \quad (2-85)$$

and for the second subsystem, $h_{d_2}(s)$ is

$$h_{d_2}(s) = \frac{8}{s^2 + 4s + 8} \quad (2-86)$$

Therefore, dynamic compensation in the form of one pole must be inserted into the control channel of subsystem 1 in order to cancel the zero and still have two subsystem poles remaining to achieve the second order response of $h_{d_1}(s)$. A dynamic compensator with only one pole must be inserted into the control channel of subsystem 2 to achieve the second order response of $h_{d_2}(s)$. Since one pole is added to each subsystem, two states of dynamic compensation are added. If $\check{x}_1(t)$ and $\check{x}_2(t)$ represent the compensator states added to subsystem 1 and subsystem 2 respectively, then the matrices $\check{A}$, $\check{B}$, $\check{C}$, and E from Eqs. (2-58) and (2-59) describing the decoupled compensation are,

$$\check{A} = \begin{bmatrix} -0.031 & 0 \\ 0 & -2 \end{bmatrix} \quad \check{B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2-87)$$

$$\check{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \check{E} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (2-87)$$

The block diagram form of the added compensation appears in Fig. 2-8b. with a pole at $s = -0.031$ added to subsystem 1 and a pole at $s = -2$. added to subsystem 2. The pole locations of the added compensation were chosen at random.

k) Determine the augmented system matrices from Eqs. (2-63), (2-64), and (2-65) for the state-variable vector definition of Eq. (2-60).

In terms of the ID system matrices, $\bar{A}$, $\bar{B}$, $\bar{C}$, and the decoupled compensation matrices $\check{A}$, $\check{B}$, $\check{C}$, and $\check{E}$, the augmented system matrices are,

$$\tilde{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0.26 & -0.031 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -0.031 & 0 \\ 0 & 0 & 0 & 0 & -2 \end{bmatrix} \quad (2-88)$$

$$\tilde{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \tilde{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

1) Repeat procedural steps a) through i) for the augmented system matrices.

Repeating step a), the scalars d_i for $i = 1, 2$ are

$$d_1 = 1, \quad d_2 = 1 \quad (2-89)$$

The computational results for the B^* , B^{*-1} , A^* , F^* , and G^* matrices appear in Fig. 2-9. The ID system representation of the augmented system matrices resulting from step e) is presented in Fig. 2-10a.

The transformation matrix T and its inverse T^{-1} , used to transform the ID system equations to the canonically decoupled (CD) representation is given in Fig. 2-10b where results of the computation indicate that $p_1 = 3$ and $p_2 = 2$. Finally, the CD system matrices from step g) namely $\hat{A}$, $\hat{B}$, and $\hat{C}$, appear in Fig. 2-11. The partitioning of the matrices in Fig. 2-11 is accomplished by computing the ℓ_i scalars for $i = 1, 2$ where repeating step h), $\ell_1 = 1$ and $\ell_2 = 0$. The partitioned form of $\hat{A}$, $\hat{B}$, and $\hat{C}$, and the internal structure of $\hat{A}_{11}$, $\hat{A}_{22}$, $\hat{B}_{11}$, $\hat{B}_{22}$, $\hat{C}_{11}$, and $\hat{C}_{22}$ agree with Figs. 2-3 and 2-4 so, therefore, the augmented system is canonically decoupled. There are three poles and one zero in subsystem 1 and only two poles in subsystem 2, since $p_1 = 3$, $\ell_1 = 1$, and $p_2 = 2$, $\ell_2 = 0$ respectively.

m) Determine the general forms for the $\hat{F}$ and $\hat{G}$ matrices using Fig. 2-5 and compute the closed-loop transfer functions for each subsystem.

$$B^* = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$B^{*-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A^* = \begin{bmatrix} 0 & 0 & 0 & -0.031 & 0 \\ 0 & 0 & 0 & 0 & -2 \end{bmatrix}$$

$$F^* = \begin{bmatrix} 0 & 0 & 0 & 0.031 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

$$G^* = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Fig. 2-9. Integrator Decoupling Matrices

$$\bar{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0.26 & -0.031 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\bar{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

a) Integrator Decoupled System Matrices

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad T^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

b) Canonical Transformation Matrices

Fig. 2-10. ID System and CD Transformation Matrices

$$\hat{A} = \left[\begin{array}{ccc|cc} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0.26 & 0 & -0.031 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right]$$

$$\hat{B} = \left[\begin{array}{c|c} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ \hline 0 & 0 \\ 0 & 1 \end{array} \right]$$

$$\hat{C} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 \end{array} \right]$$

Fig. 2-11. Canonically Decoupled System Matrices

Since $p_1 = 3$ and $p_2 = 2$, the permissible forms for $\hat{F}$ and $\hat{G}$ which preserve decoupling are

$$\hat{F} = \begin{bmatrix} f_{11} & f_{12} & f_{13} & 0 & 0 \\ 0 & 0 & 0 & f_{24} & f_{25} \end{bmatrix} \quad (2-90)$$

$$\hat{G} = \begin{bmatrix} g_{11} & 0 \\ 0 & g_{22} \end{bmatrix}$$

where

$$\hat{F}_{11} = [f_{11} \quad f_{12} \quad f_{13}] \quad \hat{F}_{22} = [f_{24} \quad f_{25}]$$

$$\hat{G}_{11} = [g_{11}] \quad \hat{G}_{22} = [g_{22}]$$

For subsystem 1, the transfer function h_{11} is

$$h_{11} = \hat{C}_{11} [sI_3 - \hat{A}_{11} - \hat{B}_{11} \hat{F}_{11}]^{-1} \hat{B}_{11} \hat{G}_{11}$$

or,

$$h_{11} = [1 \quad 0 \quad 0] \begin{bmatrix} s & -1 & 0 \\ -f_{11} & s - f_{12} & -f_{13} \\ -0.26 & 0 & s + 0.031 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ g_{11} \\ 0 \end{bmatrix}$$

$$h_{11} = \frac{g_{11}(s + 0.031)}{(s + 0.031)(s^2 - f_{12}s - f_{11}) - 0.26 f_{13}} \quad (2-91)$$

For subsystem 2, the transfer function h_{22} is,

$$h_{22} = \hat{C}_{22} [sI_2 - \hat{A}_{22} - \hat{B}_{22} \hat{F}_{22}]^{-1} \hat{B}_{22} \hat{G}_{22}$$

or,

$$h_{22} = [1 \quad 0] \begin{bmatrix} s & -1 \\ -f_{24} & s - f_{25} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ g_{22} \end{bmatrix}$$

$$h_{22} = \frac{g_{22}}{s^2 - f_{25}s - f_{24}} \quad (2-92)$$

n) Calculate the feedback and gain elements of $\hat{F}$ and $\hat{G}$ so that the transfer functions for the subsystems are equal to the desired closed-loop transfer functions from step j).

For subsystem 1, if $f_{11} = -2$, $f_{12} = -2$, $f_{13} = 0$, and $g_{11} = 2$, then

$$h_{11} = \frac{2}{s^2 + 2s + 2} = h_{d1} \quad (2-93)$$

For subsystem 2, if $f_{24} = -8$, $f_{25} = -4$, and $g_{22} = 8$, then

$$h_{22} = \frac{8}{s^2 + 4s + 8} = h_{d2} \quad (2-94)$$

and so,

$$\hat{F} = \begin{bmatrix} -2 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -8 & -4 \end{bmatrix} \quad (2-95)$$

$$\hat{G} = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}$$

o) From Eqs. (2-67) and (2-68), determine the $\tilde{F}$ and $\tilde{G}$ matrices used to control the original augmented system described by Eqs. (2-61) and (2-62).

For this example,

$$B^{*-1} \hat{F}T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -2 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -8 & -4 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

so,

$$\tilde{F} = \begin{bmatrix} 0 & 0 & 0 & 0.031 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} + \begin{bmatrix} -2 & 0 & 0 & -2 & 0 \\ 0 & 0 & -8 & 0 & -4 \end{bmatrix}$$

$$\tilde{F} = \begin{bmatrix} -2 & 0 & 0 & -1.969 & 0 \\ 0 & 0 & -8 & 0 & -2 \end{bmatrix} \quad (2-96)$$

and

$$\tilde{G} = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix} \quad (2-97)$$

p) Compute the closed-loop transfer matrix for the augmented system using the control-law of Eq. (2-66).

In general,

$$\tilde{H}(s, \tilde{F}, \tilde{G}) = \tilde{C} [sI_n - \tilde{A} - \tilde{B}\tilde{F}]^{-1} \tilde{B}\tilde{G} \quad (2-98)$$

So, for this example,

$$\tilde{H}(s, \tilde{F}, \tilde{G}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} s & 0 & 0 & -1 & 0 \\ -0.26 & s+0.031 & 0 & 0 & 0 \\ 0 & 0 & s & 0 & -1 \\ 2 & 0 & 0 & s+2 & 0 \\ 0 & 0 & 8 & 0 & s+4 \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 2 & 0 \\ 0 & 8 \end{bmatrix}$$

or,

$$\tilde{H}(s, \tilde{F}, \tilde{G}) = \begin{bmatrix} \frac{2}{s+2s+2} & 0 \\ 0 & \frac{8}{s+4s+8} \end{bmatrix} \quad (2-99)$$

where

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \underline{y}(s) = \tilde{H}(s, \tilde{F}, \tilde{G}) \begin{bmatrix} r_1(s) \\ r_2(s) \end{bmatrix} \quad (2-100)$$

This completes the design procedure since the transfer matrix of the augmented multivariable system is diagonalized and, therefore, each input affects one and only one system output. In this example, the desired second-order responses were realized exactly and represent transfer functions between the control-inputs $r_1(t)$ and $r_2(t)$ and the system outputs $y_1(t)$ and $y_2(t)$, respectively. No attempt is made to physically define the control-inputs $r_1(t)$ and $r_2(t)$ since the purpose of the example is only to illustrate the procedural steps involved in decoupling a multivariable system.

Summary

The objective of this chapter was to provide the necessary information required to design a decoupled multivariable control system using state-variable feedback.

First, the criteria to determine whether a given multivariable system is controllable was presented. Next, the question of whether or not a given multivariable system can be decoupled using state-variable feedback was answered by applying the test of non-singularity to the B matrix, a necessary and sufficient condition for decoupling proven by Falb and Wolovich (1967). The restricted pair of decoupling matrices F^* and G^* were introduced, and the shortcomings of this resultant integrator decoupled system were noted.

Gilbert's contributions included the formulation of the required transformation from the integrator decoupled form of the original multivariable system to the canonically decoupled system representation. With Gilbert's proof of a general form for the canonical system representation, the various single-input, single-output subsystems and the number of subsystem poles and zeros could consequently be identified. Furthermore, explicit forms for the feedback and gain matrices, which ensure that the resultant closed-loop system maintains the decoupled nature of the subsystems were presented.

Gilbert and Pivnichny (1969) have written a computer program which performs all the necessary computations required to decouple a multivariable system by state variable feedback. By incorporating all the decoupling results obtained by Falb and Wolovich (1967) and by

Gilbert (1969) into the computer program, the decoupling problem has been reduced to a practical synthesis procedure,

The work of Slivinsky established the procedure for adding dynamic compensators to the single-input, single-output subsystems in order to achieve a specific subsystem response using state-variable feedback. He proved that by adding compensators of a particular form to the integrator decoupled representation of the original multivariable system, the resultant canonically decoupled subsystems contain the required dynamics without an inherent loss of decoupling. The procedures involved in the design of a decoupled multivariable control system were finally illustrated with a simple example.

One important problem still is unresolved. That is, suppose the design engineer desires a closed-loop multivariable control system that possesses, in general, interaction between the various control inputs and the system outputs. The formulation of a design method by which a desired transfer matrix is realized constitutes the objective of the next chapter. The desired transfer matrix contains specific transfer functions between each control input and each and every system output.

CHAPTER 3

DESIGN OF INTERACTING MULTIVARIABLE CONTROL SYSTEMS USING DECOUPLING TECHNIQUES

This chapter presents the formulation of the feedback synthesis design technique for the realization of an interacting closed-loop multivariable control system. The technique utilizes the methods described in Chapter 2 for decoupling a multiple-input, multiple-output system into single-input, single-output subsystems using state variable feedback. The consequence of the decoupling is that the closed-loop system transfer matrix is diagonalized, and therefore each system output is affected by one and only one system input.

By incorporating cross-coupling, the transfer function matrix possesses, in general, all non-zero entries, thus each system output is affected by all system inputs in a specified desired manner. The limitations on the mathematical model are the same as those given in Chapter 1.

The computational procedure for the feedback synthesis of a coupled or interacting multivariable control system depends on the fact that for a set of n first-order differential equations, there are many possible matrix representations with the form of Eqs. (2-1) and (2-2), depending on the definitions of the vector quantities. Each representation will decompose to the same set of differential equations, but the system matrices will not necessarily obey the equivalency requirements of elementary matrix algebra.

Consider again the transfer function relationship between the input and output vectors;

$$\underline{y}(s) = H(s)\underline{r}(s) \quad (3-1)$$

or in expanded form:

$$\underline{y}(s) = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1m} \\ h_{21} & h_{22} & \cdots & h_{2m} \\ \vdots & \vdots & & \vdots \\ h_{m1} & h_{m2} & \cdots & h_{mm} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ \vdots \\ r_m \end{bmatrix} \quad (3-2)$$

The off-diagonal elements are the desired transfer function cross-coupling elements between the various m inputs and m outputs. Eq. (3-2) can equivalently be written as

$$y(s) = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix} = \left\{ \begin{bmatrix} h_{11} & 0 & \dots & 0 \\ 0 & h_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h_{mm} \end{bmatrix} \right. \quad (3-3)$$

$$+ \begin{bmatrix} 0 & 0 & \dots & 0 & h_{1m} \\ h_{21} & 0 & \dots & 0 & 0 \\ 0 & h_{32} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & h_{m,m-1} & 0 \end{bmatrix}$$

+

$$+ \left\{ \begin{bmatrix} 0 & h_{12} & 0 & \dots & 0 \\ 0 & 0 & h_{23} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_{m-1,m} \\ h_{m1} & 0 & 0 & \dots & 0 \end{bmatrix} \right\} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ \vdots \\ r_m \end{bmatrix}$$

It is noted that the first partial matrix of $H(s)$ is in diagonal or decoupled form and that each of the other component matrices can be put into diagonal form by a simple row interchange. This forms the central idea behind a systematic cross-coupling technique which will

utilize repetitive decoupling. This repetitive decoupling is called variational decoupling. The development of this together with the necessary subsequent recoupling method constitutes the main contributions of this chapter and of this extension of multiple-input, multiple-output control theory.

Variational Decoupling

To begin, $\underline{y}(s)$ is subdivided into partial output vectors, each partial vector corresponding to one of the components on the right-hand side of Eq. (3-3). That is:

$$\underline{y}(s) = \underline{y}_1(s) + \underline{y}_2(s) + \cdots + \underline{y}_m(s) \equiv \underline{y}_T(s) \quad (3-4)$$

where

$$\underline{y}_1(s) = \begin{bmatrix} y_{1_1} \\ y_{2_1} \\ y_{3_1} \\ \vdots \\ y_{m_1} \end{bmatrix} = \begin{bmatrix} h_{11} & 0 & 0 & \cdots & 0 \\ 0 & h_{22} & 0 & \cdots & 0 \\ 0 & 0 & h_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{mm} \end{bmatrix} \underline{r}(s) \quad (3-5)$$

$$\underline{y}_2(s) = \begin{bmatrix} y_{1_2} \\ y_{2_2} \\ y_{3_2} \\ \vdots \\ y_{m_2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 & h_{1m} \\ h_{21} & 0 & \cdots & 0 & 0 \\ 0 & h_{32} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & h_{m,m-1} & 0 \end{bmatrix} \underline{r}(s)$$

$$\begin{matrix} \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{matrix}$$

$$\underline{y}_m(s) = \begin{bmatrix} y_{1_m} \\ y_{2_m} \\ y_{3_m} \\ \vdots \\ y_{m_m} \end{bmatrix} = \begin{bmatrix} 0 & h_{12} & 0 & \cdots & 0 \\ 0 & 0 & h_{23} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{m-1,m} \\ h_{m,1} & 0 & 0 & \cdots & 0 \end{bmatrix} \underline{r}(s)$$

and where the subscript T on the output vector is introduced to emphasize that this vector pertains to the cross-coupled total output vector.

The subscript on the k^{th} output vector component $y_k(s)$, in addition to designating the k^{th} term on the right-hand side of Eq. (3-3), also corresponds to the output vector index of that transfer matrix element h_{ij} occurring in the first column of each of the partial transfer matrices. As will become evident, this provides a relatively convenient method of "bookkeeping" the many calculations which are required.

The idea now is to put each of the m partial transfer matrices of H in diagonal decoupled form, determine the m associated feedback and gain matrix pairs for each partial decoupled system according to the methods of Chapter 2, and finally recombine the m pairs of feedback and gain matrices which result in the closed loop transfer matrix of Eq. (3-2).

The first partial transfer matrix, associated with $y_1(s)$, is already in diagonal form, and the methods of Chapter 2 can be applied immediately. The associated feedback and gain matrices F and G for this partial system will be denoted by F_1 and G_1 respectively.

To bring the remaining partial transfer matrices to the desired diagonal form, the transformation matrix Q , defined by:

$$Q \equiv \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (3-6)$$

is utilized.

Multiplying $y_2(s)$ by Q results in:

$$Qy_2(s) = \begin{bmatrix} y_{2_2} \\ y_{3_2} \\ \vdots \\ y_{m_2} \\ y_{1_2} \end{bmatrix} = \begin{bmatrix} h_{21} & 0 & \dots & 0 & 0 \\ 0 & h_{32} & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & h_{m,m-1} & 0 \\ 0 & 0 & \dots & 0 & h_{1m} \end{bmatrix} \tilde{r}(s) \quad (3-7)$$

In general, for the k^{th} term, multiplying $y_k(s)$ by Q^{k-1} :

$$\tilde{y}_k(s) = Q^{k-1} y_k(s) = \begin{bmatrix} y_{k_k} \\ y_{k+1_k} \\ \vdots \\ y_{m_k} \\ y_{1_k} \\ \vdots \\ y_{k-1_k} \end{bmatrix} \quad (3-8)$$

where

$$\tilde{y}_k(s) = \tilde{H}_k(s, F_k, G_k) \tilde{r}(s) \quad (3-9)$$

with

$$\bar{H}_k(s, F_k, G_k) = \begin{bmatrix} h_{k,1} & & & & 0 \\ & h_{k+1,2} & & & \\ & \ddots & & & \\ & & h_{m,m-k+1} & & \\ & & & h_{1,m-k+2} & \\ & & & \ddots & \\ 0 & & & & h_{k-1,m} \end{bmatrix} \quad (3-10)$$

Thus, Eq. (3-8) defines $\bar{y}_k(s)$, the row transformed output vector and $\bar{H}_k(s, F_k, G_k)$, the diagonal partial transfer matrix. Each of the row transformed partial transfer matrices are in the required decoupling diagonal form. The techniques of Chapter 2 may now be applied to each of these decoupled transformed partial systems to determine $(m-1)$ F_k and G_k matrix pairs. Finally, the m pairs of F_k and G_k matrices are combined into a single pair of matrices F_T and G_T , where, as indicated before, the T subscript is introduced to emphasize that these are the matrices associated with the desired totally interacting system.

The procedure can equivalently be stated as follows: the original multiple-input, multiple-output system is first decoupled according to the methods of Chapter 2, and the desired feedback and gain matrices, denoted now as F_1 and G_1 , are determined. At this point, r_1 affects only y_1 , r_2 only y_2 , etc. To incorporate cross coupling, the output vector is redefined $(m-1)$ times so that each time a decoupled transfer matrix arises. In the first of these partial representations, r_1 affects only y_2 , r_2 affects only y_3 , etc., up to r_m , which affects

only y_1 . In the second partial representation, r_1 affects only y_3 , r_2 affects only y_4 , etc., again up to r_m , which affects only y_2 . This same description is repeated in a similar fashion until the $(m-1)^{th}$ partial representation is reached. Each of these m decoupled closed-loop systems consequently serves to generate m of the cross-coupling terms in the desired $m \times m$ cross-coupled transfer matrix of Eq. (3-2).

Because of the superposition principal of linear systems, the state vector $\underline{x}(t)$ can also be partitioned into separate components, each component corresponding to the partial decoupled system represented by each term of Eq. (3-4). As with $\underline{y}_T(t)$, $\underline{x}_T(t)$ will be used to denote the "total" system response vector comprised of contributions from each of the m partial responses which correspond to the m decoupled representations. Thus, the vector-matrix representation for the interacting closed-loop system is:

$$\dot{\underline{x}}_T(t) = A\underline{x}_T(t) + B\underline{u}_T(t) \quad (3-11)$$

$$\underline{y}_T(t) = C\underline{x}_T(t) \quad (3-12)$$

$$\underline{u}_T(t) = F_T\underline{x}_T(t) + G_T\underline{r}(t) \quad (3-13)$$

with

$$\underline{x}_T(t) = \sum_{k=1}^m \underline{x}_k(t) \quad (3-14)$$

$$\underline{y}_T(t) = \sum_{k=1}^m \underline{y}_k(t) = C \sum_{k=1}^m \underline{x}_k(t) \quad (3-15)$$

The k^{th} decoupled system satisfies the equations:

$$\dot{\tilde{x}}_k(t) = A\tilde{x}_k(t) + B\tilde{u}_k(t) \quad (3-16)$$

$$\tilde{y}_k(t) = Q^{k-1}y_k(t) = Q^{k-1}C\tilde{x}_k(t) = C_k\tilde{x}_k(t) \quad (3-17)$$

$$\tilde{u}_k(t) = F_k\tilde{x}_k(t) + G_k\tilde{r}(t) \quad (3-18)$$

where

$$C_k = Q^{k-1}C, \quad k = 1, 2, \dots, m. \quad (3-19)$$

The relationship between F_k and G_k and the transfer matrix for the k^{th} decoupled vector-matrix representation is obtained by substituting Eq. (3-18) into Eq. (3-16):

$$\dot{\tilde{x}}_k(t) = (A + BF_k)\tilde{x}_k(t) + BG_k\tilde{r}(t) \quad (3-20)$$

Taking the Laplace transform of Eq. (3-20), assuming zero initial conditions on the state vector $\tilde{x}_k(t)$, and solving for $\tilde{x}_k(s)$:

$$\tilde{x}_k(s) = [sI_n - A - BF_k]^{-1}BG_k\tilde{r}(s) \quad (3-21)$$

Then using the Laplace transform of Eq. (3-17) together with Eq. (3-21):

$$\tilde{y}_k(s) = C_k\tilde{x}_k(s) = C_k[sI_n - A - BF_k]^{-1}BG_k\tilde{r}(s) \quad (3-22)$$

or, the transfer matrix, $\bar{H}_k(s, F_k, G_k)$, is

$$\bar{H}_k(s, F_k, G_k) = C_k[sI_n - A - BF_k]^{-1}BG_k \quad (3-23)$$

The bar over $H_k(s, F_k, G_k)$ is used to emphasize that the transfer matrix is in diagonal form.

The elements of $\bar{H}_k(s, F_k, G_k)$ are identified from the desired transfer matrix [see Eq. (3-10)] and the F_k and G_k matrices are then determined using the decoupling techniques of Chapter 2. This process is simply repeated m times until m pairs of F_k and G_k matrices are determined.

The total system feedback and gain matrices, F_T and G_T , are then determined in terms of the m pairs of F_k and G_k matrices by utilizing the definitions of Eq. (3-14) and (3-15) while satisfying Eqs. (3-11), (3-12), and (3-13).

The question may arise at this point whether or not each of the vector-matrix representations used in the variational decoupling technique can be decoupled or, equivalently, if the B^* matrix is singular for any of the partial vector-matrix representations. The theorem below proves that if the original vector-matrix representation can be decoupled, then performing the elementary row operations on the original vector-matrix representation does not cause a loss of decoupling for any of the partial vector-matrix representations.

Theorem 3.1

If B^* is nonsingular for the original vector-matrix representation, it is also nonsingular for all partial vector-matrix representations generated with the transformation matrix Q of Eq. (3-6).

Proof:

Define B_1^* to be the matrix which tests the original vector-matrix representation for decouplability. Then, using Eq. (2-17),

$$B_1^* = \begin{bmatrix} C_{11} A^{d_{11} B} \\ C_{12} A^{d_{12} B} \\ \vdots \\ C_{1m} A^{d_{1m} B} \end{bmatrix} \quad (3-24)$$

where C_{1i} denotes the i^{th} row of the original output matrix, and using Eq. (2-14),

$$d_{1i} = \min [j, C_{1i} A^j B \neq 0, \quad j = 0, 1, \dots, n-1] \quad (3-25)$$

The B^* matrix for any k^{th} partial vector-matrix representation is denoted B_k^* , where again using Eq. (2-17),

$$B_k^* = \begin{bmatrix} C_{k1} A^{d_{k1} B} \\ C_{k2} A^{d_{k2} B} \\ \vdots \\ C_{km} A^{d_{km} B} \end{bmatrix} \quad (3-26)$$

and C_{ki} denotes the i^{th} row of the k^{th} partial output matrix, and again using Eq. (2-14),

$$d_{ki} = \min [j; C_{ki} A^j B \neq 0, \quad j = 0, 1, \dots, n-1] \quad (3-27)$$

where all the C_k , including $C_1 = C$, are given by Eq. (3-19). The i^{th} row of the output matrix C_k for the k^{th} partial decoupled system is

related to the j^{th} row of the original system output matrix C_1 by:

$$C_{k_i} = C_{1_j} \quad (3-28)$$

where $j = i+k-1$ for $i \leq m-k+1$ or $j = i+k-m-1$ for $m-k+1 \leq i \leq m$.

Therefore, using Eq. (3-25) to relate the d_{k_i} scalars in terms of the d_{1_i} scalars for each C_{k_i} row matrix,

$$B_k^* = \begin{bmatrix} C_{1_k} & A^{d_{1_k}} & B \\ & \vdots & \\ C_{1_m} & A^{d_{1_m}} & B \\ & \vdots & \\ C_{1_1} & A^{d_{1_1}} & B \\ & \vdots & \\ C_{1_{k-1}} & A^{d_{1_{k-1}}} & B \end{bmatrix} \quad (3-29)$$

or, from Eq. (3-24),

$$B_k^* = Q^{k-1} B_1^* \quad (3-30)$$

Since the inverse of a product is the product of the matrix inverses in reverse order,

$$B_k^{*-1} = [Q^{k-1} B_1^*]^{-1} = B_1^{*-1} [Q^{k-1}]^{-1} \quad (3-31)$$

and since the inverse of Q exists, the inverse of Q to the power $(k-1)$ also exists. Because it is hypothesized that B_1^* is nonsingular, the inverse of B_k^* does exist, and therefore the k^{th} partial vector-matrix

representation can be decoupled. Allowing k to range from 2 to m completes the proof.

(End Theorem 3.1)

The repeated computation of F^* for each vector matrix representation also is not required since F^* is invariant to the elementary row transformations performed by Q . This is proven in Theorem 3.2 below:

Theorem 3.2

The F^* matrix which integrator decouples a multivariable system is invariant for any vector-matrix representation generated with the transformation matrix Q from Eq. (3-6).

Proof:

The F_1^* matrix for the original vector-matrix representation is:

$$F_1^* = -B_1^{*-1} A_1^* \quad (3-32)$$

where B_1^* is given by Eq. (3-24) and,

$$A_1^* = \begin{bmatrix} C_{11} A^{d_{11}+1} \\ C_{12} A^{d_{12}+1} \\ \vdots \\ C_{1m} A^{d_{1m}+1} \end{bmatrix} \quad (3-33)$$

Likewise, the F_k^* matrix for any k^{th} partial vector-matrix representation, $k = 2, 3, \dots, m$, is:

$$F_k^* = -B_k^{*-1} A_k^* \quad (3-34)$$

where B_k^* is given by Eq. (3-26) or (3-30) and

$$A_k^* = \begin{bmatrix} C_{k_1} A_{k_1}^{d_{k_1}+1} \\ C_{k_2} A_{k_2}^{d_{k_2}+1} \\ \vdots \\ C_{k_m} A_{k_m}^{d_{k_m}+1} \end{bmatrix} \quad (3-35)$$

or, as was the case for B_k^* :

$$A_k^* = Q^{k-1} A_1^* \quad (3-36)$$

Substituting Eq. (3-31) for B_k^{*-1} , the result is

$$F_k^* = -B_1^{*-1} [Q^{k-1}]^{-1} Q^{k-1} A_1^* \quad (3-37)$$

or

$$F_k^* = -B_1^{*-1} A_1^* = F_1^* \quad (3-38)$$

Again allowing k to range from 2 to m completes the proof.

(End Theorem 3.2)

From Theorem 3.1, it is not necessary to determine B_k^* for any of the partial vector-matrix representations. Therefore, the expression for the gain matrix G_k^* can be obtained in terms of the B_1^* matrix from the original vector-matrix notation. This formulation results

immediately since, corresponding to Eq. (2-20) (letting $L = I_m$, the $m \times m$ identity matrix):

$$G_k^* = B_k^{*-1} \quad (3-39)$$

and, using Eq. (3-31),

$$G_k^* = G_1^* [Q^{k-1}]^{-1} \quad (3-40)$$

where

$$G_1^* = B_1^{*-1} \quad (3-41)$$

The original and each partial vector-matrix representation can now be integrator decoupled using Eqs. (3-24), (3-33), (3-32), (3-41), and (3-40) for B_1^* , A_1^* , $F_1^* = F_k^*$, G_1^* , and G_k^* , respectively.

Since the A and B system matrices do not change from one vector-matrix representation to the next, the formulas of Eqs. (2-34) and (2-35), applicable to $\bar{A}$ and $\bar{B}$, require no additional notation to indicate dependence on a particular vector-matrix representation. However, since the output matrix is different for each representation, the formula for $\bar{C}$ [Eq. (2-36)] becomes

$$\bar{C}_k = C_k \quad (3-42)$$

for the k^{th} vector-matrix representation.

The first d_i+1 rows, $i = 1, 2, \dots, m$, of each submatrix T_i of the transformation matrix in Eq. (2-50) depend on the system output matrix. Since each output matrix C_k is different for $k = 1, 2, \dots, m$, in general, each transformation matrix T_k will be different for different vector-matrix notations. In this chapter, T_k will be used to denote the

transformation matrix associated with the k^{th} partial decoupled vector-matrix representation. This should not be confused with the T_i notation of Chapter 2 used to indicate the row submatrices of T [Eq. (2-50)].

Even though the decoupled form for the canonically decoupled system must be the same for each vector-matrix representation, each $\hat{A}$, $\hat{B}$, and $\hat{C}$ will in general be different due to the different transformation matrices T_k . Therefore, in the notation of this chapter,

$$\hat{A}_k = T_k (A + BF_1^*) T_k^{-1} \quad (3-43)$$

$$\hat{B}_k = T_k B G_k^* \quad (3-44)$$

$$\hat{C}_k = C T_k^{-1} \quad (3-45)$$

where each $\hat{A}_k$, $\hat{B}_k$, and $\hat{C}_k$, $k = 1, 2, \dots, m$ must have the form shown in Fig. (2-3).

Finally, in the canonically decoupled form, each subsystem of each partial vector-matrix representation possesses a different single-input, single-output relationship. The $\hat{F}$ and $\hat{G}$ matrices will consequently differ generally from one vector-matrix representation to the next. Therefore, the particular feedback and gain matrices in the canonical system which produce the desired transfer functions between an element in the output vector $\bar{y}_k(t)$ and the corresponding elements in the external control input vector $\bar{r}(t)$, are denoted $\hat{F}_k$ and $\hat{G}_k$, $k = 1, 2, \dots, m$. In terms of the original system matrices for the k^{th} vector-matrix representation, Eqs. (2-56) and (2-57) become:

$$F_k = F_1 + B_k^{*-1} \hat{F}_k T_k \quad (3-46)$$

and

$$G_k = B_k^{*-1} \hat{G}_k \quad (3-46)$$

where B_k^{*-1} is given in terms of B_1^{*-1} by Eq. (3-31).

Formulation of Cross-Coupling Control-Law Matrices

There remains the task of formulating the single pair of "total" feedback and gain matrices, F_T and G_T , expressed in terms of the $\underline{m}$ F_k and G_k partial matrix pairs. The F_T and G_T matrices then comprise the cross-coupled feedback control law for the original system according to Eq. (3-13).

From Eqs. (3-11) and (3-13),

$$\dot{\tilde{x}}_T(t) = (A + BF_T)\tilde{x}_T(t) + BG_T\tilde{r}(t) \quad (3-48)$$

Similarly, from Eqs. (3-16) and (3-18):

$$\dot{\tilde{x}}_k(t) = (A + BF_k)\tilde{x}_k(t) + BG_k\tilde{r}(t) \quad (3-49)$$

Taking the time derivative of Eq. (3-14) and incorporating this with Eqs. (3-48) and (3-49) above results in:

$$(A + BF_T)\tilde{x}_T(t) + BG_T\tilde{r}(t) = \sum_{k=1}^m (A + BF_k)\tilde{x}_k(t) + B \sum_{k=1}^m G_k\tilde{r}(t) \quad (3-50)$$

Since this relationship must hold for arbitrary $\tilde{r}(t)$ (including the null vector), G_T must be defined as

$$G_T = \sum_{k=1}^m G_k \quad (3-51)$$

Substituting Eq. (3-51) into Eq. (3-50), cancelling $A\mathbf{x}_T$ by using Eq. (3-14) and premultiplying the result by the left-inverse of B, results in:

$$\mathbf{F}_T \mathbf{x}_T(t) = \sum_{k=1}^m \mathbf{F}_k \mathbf{x}_k(t) \quad (3-52)$$

Introducing Eq. (3-14) for $\mathbf{x}_T(t)$, taking the Laplace transform, and then substituting Eq. (3-21) for $\mathbf{x}_k(s)$ results in

$$\mathbf{F}_T \sum_{k=1}^m [\mathbf{sI}_n - \mathbf{A} - \mathbf{BF}_k]^{-1} \mathbf{BG}_k = \sum_{k=1}^m \mathbf{F}_k [\mathbf{sI}_n - \mathbf{A} - \mathbf{BF}_k]^{-1} \mathbf{BG}_k \quad (3-53)$$

Defining

$$\mathbf{D}_k(s) = [\mathbf{sI}_n - \mathbf{A} - \mathbf{BF}_k]^{-1} \mathbf{BG}_k \quad (3-54)$$

Eq. (3-53) becomes:

$$\mathbf{F}_T \sum_{k=1}^m \mathbf{D}_k(s) = \sum_{k=1}^m \mathbf{F}_k \mathbf{D}_k(s) \quad (3-55)$$

Eqs. (3-51) and (3-55) relate the desired feedback and gain matrices, $\mathbf{F}_T$ and $\mathbf{G}_T$, to the set of m decoupling matrices $\mathbf{F}_k$ and $\mathbf{G}_k$, $k = 1, 2, \dots, m$. Although $\mathbf{G}_T$ can be computed directly from the $\mathbf{G}_k$ matrices, no general closed-form solution for $\mathbf{F}_T$ exists because the matrix represented by $\sum_{k=1}^m \mathbf{D}_k$ is an $n \times m$ matrix, and therefore neither the inverse nor the right-inverse is defined for $m < n$. However, a procedure for obtaining an $\mathbf{F}_T$ matrix which satisfies the relationship of Eq. (3-55) does exist and will be presented subsequently.

An important point concerning Eq. (3-55) should perhaps be clarified before proceeding. Except for the special case of all F_k matrices equal, F_T may contain an $\underline{s}$ dependence and hence dynamic elements. Therefore, the Laplace transform of Eq. (3-52) would not result in Eq. (3-53) because the transform of the product of the two time functions is not a simple product of the two functions in the Laplace domain. However, if Eq. (3-55) is assumed to express the correct and desired relationship for F_T , then the control law must be interpreted in the more general representation:

$$u_T(t) = F_T(t) * x_T(t) + G_T r(t) \quad (3-56)$$

where the symbol (*) means the convolution integral of the product $F_T(t)$ and $x_T(t)$. The previous control-law, Eq. (3-13), is just a special case of the more general form of control-law given by Eq. (3-56). Eq. (3-13) arises from assuming constant feedback elements in $F_T(s)$, consequently $F_T(t)$ is $F_T \delta(t)$ (where $\delta(t)$ is the Dirac delta function) and Eq. (3-56) reduces to Eq. (3-13).

Computation of D_k Matrices

The necessity to discuss the computation of the D_k matrices of Eq. (3-54) exists because each computation requires the inversion of $[sI_n - A - BF_k]$ for $k = 1, 2, \dots, m$. For orders of $n > 4$, the inversion of a large matrix containing polynomials with the Laplace variable $\underline{s}$ is a difficult and time-consuming operation. Yet a sufficiently fast and efficient computer code does not appear to have been programmed to date (Freeman and Abbott, 1969) to facilitate the inversion of a large matrix

containing polynomial elements of an unspecified variable. An attempt is made to rectify this situation by examining the general form of the D_k matrices. The method of matrix partitioning described below is then used to compute the inverses.

Since the general form of the canonically decoupled system matrices is known, the structure of the D_k matrices can be determined by deriving Eq. (3-54) in terms of the $\hat{A}_k$, $\hat{B}_k$, $\hat{F}_k$, and $\hat{G}_k$ canonical system matrices.

Solve Eqs. (3-43) and (3-44) for $(A + BF_1^*)$ and B , using Eq. (3-39) for G_k^* :

$$A + BF_1^* = T_k^{-1} \hat{A}_k T_k \quad (3-57)$$

$$B = T_k^{-1} \hat{B}_k B_k^* \quad (3-58)$$

Substitute Eqs. (3-46) and (3-47) into Eqs. (3-54) to get:

$$D_k = [sI_n - A - BF_1^* - BB_k^{*-1} \hat{F}_k T_k]^{-1} BB_k^{*-1} \hat{G}_k \quad (3-59)$$

Substitute Eqs. (3-57) and (3-58) into Eq. (3-59), then

$$D_k = [sI_n - T_k^{-1} \hat{A}_k T_k - T_k^{-1} \hat{B}_k \hat{F}_k T_k]^{-1} T_k^{-1} \hat{B}_k \hat{G}_k \quad (3-60)$$

With a little matrix algebra, Eq. (3-60) can be written as:

$$D_k = T_k^{-1} [sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]^{-1} \hat{B}_k \hat{G}_k \quad (3-61)$$

The computation of the D_k terms now consists of finding the inverse of $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]$. By using the general form or structure for the $\hat{A}_k$, $\hat{B}_k$, and $\hat{F}_k$ matrices given in Figs. 2-3, 2-4, and 2-5, the

general form of $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]$ results in the R matrix,

$$R = \begin{bmatrix} R_{11} & | & R_{12} \\ \hline R_{21} & | & R_{22} \end{bmatrix} \quad (3-62)$$

as illustrated in Fig. 3-1.

A formula for the inverse of the matrix R is given by Frame (1964) without proof as:

$$R^{-1} = \begin{bmatrix} I & -R_{11}^{-1}R_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} R_{11}^{-1} & 0 \\ 0 & (R_{22} - R_{21}R_{11}^{-1}R_{12})^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -R_{21}R_{11}^{-1} & I \end{bmatrix} \quad (3-63)$$

where I represents the identity matrix of appropriate order. One advantage for using this form for the inverse of a matrix is that the inversion requires the calculation of inverses of smaller submatrices.

Denoting $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]^{-1}$ as R^{-1} , and since R_{12} is a null matrix, Eq. (3-63) becomes:

$$[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]^{-1} = \begin{bmatrix} R_{11}^{-1} & 0 \\ -R_{22}^{-1}R_{21}R_{11}^{-1} & R_{22}^{-1} \end{bmatrix} \quad (3-64)$$

Therefore, if the inverse of $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]$ is computed using Eq. (3-64) with the partitioned matrices as shown in Fig. 3-1, the required calculation is simplified to taking the inverses of $m+1$ reduced matrices, namely $[sI_{k_{ii}} - \hat{A}_{k_{ii}} - \hat{B}_{k_{ii}} \hat{F}_{k_{ii}}]^{-1}$, for $i = 1, 2, \dots, m$ and

$$[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k] = \begin{bmatrix} \frac{R_{11}}{\frac{R_{21}}{R_{22}}} \vdots \frac{R_{12}}{R_{22}} \\ \vdots \end{bmatrix}_{n \times n}$$

where

$$R_{11} = \begin{bmatrix} [sI - \hat{A}_{k_{11}} - \hat{B}_{k_{11}} \hat{F}_{k_{11}}] & 0 & \cdots & \cdots & 0 \\ 0 & [sI - \hat{A}_{k_{22}} - \hat{B}_{k_{22}} \hat{F}_{k_{22}}] & \cdots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & [sI - \hat{A}_{k_{mm}} - \hat{B}_{k_m} \hat{F}_{k_{mm}}] \end{bmatrix}$$

$$R_{21} = \begin{bmatrix} -[\hat{A}_{k_1}^C + \hat{B}_{k_1}^C \hat{F}_{k_{11}}] & -[\hat{A}_{k_2}^C + \hat{B}_{k_2}^C \hat{F}_{k_{22}}] & \cdots & \cdots & -[\hat{A}_{k_m}^C + \hat{B}_{k_m}^C \hat{F}_{k_{mm}}] \end{bmatrix}$$

$$R_{22} = [sI - \hat{A}_{k_{m+1}}]$$

$R_{12} = 0$, the $n \times p_{m+1}$ null matrix

Fig. 3-1. Canonical Decoupled Structure of $[sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]$

$[sI - \hat{A}_{k_{m+1}}]^{-1}$. A computer program to compute the D_k matrices for $k = 1, 2, \dots, m$ in terms of the canonically decoupled system matrices appears in the appendix of this study.

Computational Procedures for Control-Law Realization

This section concerns the actual computation of the F_T and G_T control-law matrices which are used to obtain the desired closed-loop system response. As previously noted, the need to discuss the actual computation exists because the closed-form solution of F_T in terms of the F_k and G_k matrices cannot be formulated for $m < n$. Again, the reason is that for $m < n$, the matrix $\sum_{k=1}^m D_k$ is not square, and therefore the general inverse of the matrix cannot be defined. Also, since $\sum_{k=1}^m D_k$ is an $n \times m$ matrix, the number of columns in the matrix is less than the number of rows. Consequently, the right-inverse of $\sum_{k=1}^m D_k$ also does not exist (Hohn, 1966).

However, it is possible to compute the F_T matrix by applying the Gaussian elimination process (Hohn, 1966) to the set of m equations expressed by Eq. (3-55) which transforms the set of equations to row echelon form. By equating the coefficients of like powers of s on both sides of each of the m equations, a set of mn equations containing the mn unknown elements of F_T is obtained. The set of mn equations results because each D_k matrix for $k = 1, 2, \dots, m$ must contain s to the n^{th} power [see Eq. (3-54)].

The computation of the F_T matrix using the Gaussian elimination process is best illustrated by an example. The example used previously

in Chapter 2 to illustrate the decoupling techniques will be used for this purpose.

For completeness and clarity, the step-by-step computational procedure for the entire design process is first given and is carried out on the example below:

1. Form the original vector-matrix representation of Eqs. (3-16) and (3-17) with $k=1$ from the multivariable system differential equations using any one set of vector definitions.
2. Compute B_1^* using Eq. (3-24) and test for nonsingularity to ensure that the vector-matrix representation can be decoupled.
3. Generate the $(m-1)$ partial system vector-matrix representations of Eqs. (3-16) and (3-17) with $k = 2, 3, \dots, m$ from the original vector-matrix representation using the transformation matrix Q defined by Eq. (3-6).
4. Specify the desired closed-loop system transfer matrix by choosing the desired transfer functions between each external control input and each and every system output. Use Eq. (3-10) to identify each $\bar{H}_k(s, F_k, G_k)$ for $k = 1, 2, \dots, m$.
5. Integrator decouple the original and each partial vector-matrix representation by computing F_1^* , G_1^* , and G_k^* using Eqs. (3-32), (3-41), and (3-40), respectively.
6. Compute the transformation matrices T_k , $k = 1, 2, \dots, m$, required to transform each integrator decoupled system to the canonically decoupled system.

7. Determine the canonically decoupled system matrices $\hat{A}_k$, $\hat{B}_k$, and $\hat{C}_k$ for $k = 1, 2, \dots, m$, using Eqs. (3-43), (3-44), and (3-45), respectively.
8. Identify the subsystems for each vector-matrix representation using Eq. (2-46) and Fig. (2-3) and determine the number of poles and zeros in each subsystem.
9. Add required dynamic compensation in the form of lead-lag compensators to the control-input channels where necessary in order that the number of subsystem poles and zeros are adequate to achieve the desired subsystem transfer functions of Eq. (3-3) with state variable feedback.
10. If no dynamic compensation is required, go to Step 13; otherwise determine the augmented system matrices, $\tilde{A}$ and $\tilde{B}$, using Eqs. (2-63) and (2-64).
11. Determine the augmented system output matrix for the original and each partial system vector-matrix representation using Eq. (3-65) below,

$$\tilde{C}_k = [\bar{C}_k \quad 0] \quad (3-65)$$

where $\bar{C}_k$ is given by Eq. (3-42) and $\underline{0}$ is the $m \times n_c$ null matrix with n_c defined as the number of states resulting from the added compensation [Eq. (2-60)].

12. Repeat Steps 5 through 8 with augmented system matrices $\tilde{A}$, $\tilde{B}$, and $\tilde{C}_k$ for $k = 1, 2, \dots, m$.

13. Apply state-variable feedback to the resultant subsystems for the original and each partial vector-matrix representation and compute the $\hat{F}_k$ and $\hat{G}_k$ matrices, $k = 1, 2, \dots, m$, causing each subsystem to possess the response specified in Step 4.
14. Compute the F_k and G_k matrices from the $\hat{F}_k$ and $\hat{G}_k$ matrices using Eqs. (3-46) and (3-47) for $k = 1, 2, \dots, m$.
15. Compute each D_k matrix for $k = 1, 2, \dots, m$ using Eq. (3-61).
16. Form each $F_k D_k$ matrix using the results of Steps 14 and 15 and compute the $\sum_{k=1}^m F_k D_k$ matrix.
17. Represent F_T as a general matrix. With elements f_{ij} , $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$, form the product $F_T \sum_{k=1}^m D_k$, and equate the elements resulting from the relationship of Eq. (3-55).
18. Apply the Gaussian elimination process to the m^2 algebraic equations resulting from Step 17 to transform the equations to row echelon form.
19. Equate the coefficients of like powers of s and solve the resulting set of mn equations for the unknown elements in F_T .

Example 3.1

The differential equations describing the multivariable system for this example are those which result in the augmented system matrices of Eq. (2-88), or

$$\begin{aligned}
 \dot{x}_1(t) &= x_4(t) & y_1(t) &= x_1(t) \\
 \dot{x}_2(t) &= 0.26 x_1(t) - 0.031 x_2(t) & y_2(t) &= x_3(t) \\
 \dot{x}_3(t) &= x_5(t) \\
 \dot{x}_4(t) &= -0.031 x_4(t) + u_1(t) \\
 \dot{x}_5(t) &= -2 x_5(t) + u_2(t)
 \end{aligned} \tag{3-66}$$

The computational steps indicated above are now performed on the system differential equations of Eq. (3-66).

Step 1) Form the original vector-matrix representation by defining the system vectors as:

$$\begin{aligned}
 \underline{y}_1(t) &= \begin{bmatrix} y_{1_1}(t) \\ y_{2_1}(t) \end{bmatrix} & \underline{u}_1(t) &= \begin{bmatrix} u_{1_1}(t) \\ u_{2_1}(t) \end{bmatrix} & \underline{x}_1(t) &= \begin{bmatrix} x_{1_1}(t) \\ x_{2_1}(t) \\ x_{3_1}(t) \\ x_{4_1}(t) \\ x_{5_1}(t) \end{bmatrix}
 \end{aligned} \tag{3-67}$$

so,

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0.26 & -0.031 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -0.031 & 0 \\ 0 & 0 & 0 & 0 & -2 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3-68)$$

and

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (3-69)$$

Step 2) Using the system matrices of Eqs. (3-68) and (3-69), B_1^* and B_1^{*-1} are

$$B_1^* = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad B_1^{*-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3-70)$$

So the original vector-matrix representation can be decoupled.

Step 3) Since $m=2$, the transformation matrix of Eq. (3-6) is

$$Q = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (3-71)$$

and the only partial system vector-matrix representation is

$$\bar{y}_2(t) = Qy_2(t) = \begin{bmatrix} y_{2_2}(t) \\ y_{1_2}(t) \end{bmatrix} \quad (3-72)$$

$$u_2(t) = \begin{bmatrix} u_{1_2}(t) \\ u_{2_2}(t) \end{bmatrix} \quad x_2(t) = \begin{bmatrix} x_{1_2}(t) \\ x_{2_2}(t) \\ x_{3_2}(t) \\ x_{4_2}(t) \\ x_{5_2}(t) \end{bmatrix}$$

with A and B given by Eq. (3-68) and

$$C_2 = QC_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3-73)$$

Step 4) Assume that the desired closed-loop system transfer matrix is given as

$$H_d(s, F_T, G_T) = \begin{bmatrix} \frac{2}{s^2 + 2s + 2} & \frac{4}{s^2 + 4s + 4} \\ \frac{0}{s^2 + 4s + 3} & \frac{8}{s^2 + 4s + 8} \end{bmatrix} \quad (3-74)$$

This transfer matrix was selected on the basis that the important processes in the computation of F_T and G_T can be illustrated without undue complexity. In the transfer matrix specified by Eq. (3-74), note that a perturbation at input $r_1(t)$ will not be detected at the

system output denoted $y_2(t)$. If this element had been chosen non-zero, only the complexity of the problem would have been increased but the technique itself would be unaltered.

From Eq. (3-10),

$$\bar{H}_1(s, F_1, G_1) = \begin{bmatrix} \frac{2}{s^2 + 2s + 2} & 0 \\ 0 & \frac{8}{s^2 + 4s + 8} \end{bmatrix} \quad (3-75)$$

and

$$\bar{H}_2(s, F_2, G_2) = \begin{bmatrix} \frac{0}{s^2 + 4s + 3} & 0 \\ 0 & \frac{4}{s^2 + 4s + 4} \end{bmatrix} \quad (3-76)$$

Step 5) The F_1^* , G_1^* , and G_2^* matrices are given in Fig. 3-2.

Step 6) The transformation matrices T_1 and T_2 are given in Fig. 3-3.

Steps 7 and 8) The canonically decoupled system matrices, $\hat{A}_1$, $\hat{B}_1$, and $\hat{C}_1$ and $\hat{A}_2$, $\hat{B}_2$, and $\hat{C}_2$ are given in Fig. 3-4 together with the respective integers p_k , d_k , l_k .

Step 9) Since subsystems 1 and 2 of the original vector-matrix representation possess one zero and three poles, and no zeros and two poles, respectively, and since subsystems 1 and 2 of the partial system vector-matrix representation contain no zeros and two poles, and one zero and three poles, respectively, no series dynamic compensation is required to realize the desired transfer functions with state variable feedback.

$$F_1^* = \begin{bmatrix} 0 & 0 & 0 & 0.031 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

$$G_1^* = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$G_2^* = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Fig. 3-2. Original and Partial System Integrator Decoupling Matrices for Example 3.1

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad T_1^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} = T_2^{-1}$$

Fig. 3-3. Integrator-to-Canonically-Decoupled System Transformation Matrices for Example 3.1

$$\hat{A}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0.26 & 0 & -0.031 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \hat{B}_1 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\hat{C}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{aligned} p_{1_1} &= 3, d_{1_1} = 1, \ell_{1_1} = 1 \\ p_{2_1} &= 2, d_{2_1} = 1, \ell_{2_1} = 0 \end{aligned}$$

$$\hat{A}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.26 & 0 & -0.031 \end{bmatrix} \quad \hat{B}_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\hat{C}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad \begin{aligned} p_{1_2} &= 2, d_{1_2} = 1, \ell_{1_2} = 0 \\ p_{2_2} &= 3, d_{2_2} = 1, \ell_{2_2} = 1 \end{aligned}$$

Fig. 3-4. Canonically Decoupled System Matrices for Example 3.1

Step 13) The $\hat{F}_1, \hat{G}_1, \hat{F}_2, \hat{G}_2$ matrices are given in Fig. 3-5.

Step 14) The feedback and gain matrices $F_1, G_1, F_2,$ and G_2 , corresponding to the $\hat{F}_1, \hat{G}_1, \hat{F}_2,$ and $\hat{G}_2$ matrices from Step 13 appear in Fig. 3-6.

Step 15) The results of the D_k computations are given in Fig. 3-7.

Step 16) The $F_k D_k$ matrices are given in Fig. 3-8 along with the matrix sum, $\sum_{k=1}^m F_k D_k$.

Step 17) Since F_T must be a (2×5) matrix, let F_T be represented by:

$$F_T = \begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{14} & f_{15} \\ f_{21} & f_{22} & f_{23} & f_{24} & f_{25} \end{bmatrix} \quad (3-77)$$

Forming the product $F_T \sum_{k=1}^2 D_k$, and equating the elements in the two matrices $F_T \sum_{k=1}^2 D_k$ and $\sum_{k=1}^2 F_k D_k$, the four algebraic equations of Fig. 3-9 are obtained.

Step 18) The first two equations contain the unknowns in the first row of the general F_T matrix of Eq. (3-77). The last two equations contain the unknown elements for the second row of F_T . The Gaussian elimination process involves first clearing the four equations by the coefficients of the first terms in each equation. With a little additional algebra, the row echelon form shown in Fig. 3-10 results.

Step 19) Equating the coefficients of like powers of s in each of the equations in Fig. 3-10 and solving for the f_{ij} results in:

$$\hat{F}_1 = \begin{bmatrix} -2 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -8 & -4 \end{bmatrix} \quad \hat{G}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}$$

$$\hat{F}_2 = \begin{bmatrix} -3 & -4 & 0 & 0 & 0 \\ 0 & 0 & -4 & -4 & 0 \end{bmatrix} \quad \hat{G}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix}$$

Fig. 3-5. Canonical System Feedback and Gain Matrices for Example 3.1

$$F_1 = \begin{bmatrix} -2 & 0 & 0 & -1.969 & 0 \\ 0 & 0 & -8 & 0 & -2 \end{bmatrix} \quad G_1 = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} -4 & 0 & 0 & -3.969 & 0 \\ 0 & 0 & -3 & 0 & -2 \end{bmatrix} \quad G_2 = \begin{bmatrix} 0 & 4 \\ 0 & 0 \end{bmatrix}$$

Fig. 3-6. Original System Feedback and Gain Matrices for Example 3.1

$$D_1 = \begin{bmatrix} \frac{2}{s^2+2s+2} & 0 \\ \frac{0.52}{(s^2+2s+2)(s+0.031)} & 0 \\ 0 & \frac{8}{s^2+4s+8} \\ \frac{2s}{s^2+2s+2} & 0 \\ 0 & \frac{8s}{s^2+4s+8} \end{bmatrix}$$

$$D_2 = \begin{bmatrix} 0 & \frac{4}{s^2+4s+4} \\ 0 & \frac{1.04}{(s^2+4s+4)(s+0.031)} \\ 0 & 0 \\ 0 & \frac{4s}{s^2+4s+4} \\ 0 & 0 \end{bmatrix}$$

Fig. 3-7. D_k Matrices for Example 3.1

$$F_1 D_1 = \begin{bmatrix} \frac{-4 - 2(1.969)s}{s^2 + 2s + 2} & 0 \\ 0 & \frac{-64 - 16s}{s^2 + 4s + 8} \end{bmatrix}$$

$$F_2 D_2 = \begin{bmatrix} 0 & \frac{-16 - 4(3.969)s}{s^2 + 4s + 4} \\ 0 & 0 \end{bmatrix}$$

$$\sum_{k=1}^2 F_k D_k = \begin{bmatrix} \frac{-4 - 2(1.969)s}{s^2 + 2s + 2} & \frac{-16 - 4(3.969)s}{s^2 + 4s + 4} \\ 0 & \frac{-64 - 16s}{s^2 + 4s + 8} \end{bmatrix}$$

Fig. 3-8. $F_k D_k$ Matrices for Example 3.1

$$\frac{2f_{11}}{s^2+2s+2} + \frac{0.52f_{12}}{(s^2+2s+2)(s+0.031)} + \frac{2sf_{14}}{s^2+2s+2} = \frac{-4-2(1.969)s}{s^2+2s+2}$$

$$\frac{4f_{11}}{s^2+4s+4} + \frac{1.04f_{12}}{(s^2+4s+4)(s+0.031)} + \frac{8f_{13}}{s^2+4s+8} + \frac{4sf_{14}}{s^2+4s+4} + \frac{8sf_{15}}{s^2+4s+8} = \frac{-16-4(3.969)s}{s^2+4s+4}$$

$$\frac{2f_{21}}{s^2+2s+2} + \frac{0.52f_{22}}{(s^2+2s+2)(s+0.031)} + \frac{2sf_{24}}{s^2+2s+2} = 0$$

$$\frac{4f_{21}}{s^2+4s+4} + \frac{1.04f_{22}}{(s^2+4s+4)(s+0.031)} + \frac{8f_{23}}{s^2+4s+8} + \frac{4sf_{24}}{s^2+4s+4} + \frac{8sf_{25}}{s^2+4s+8} = \frac{-64-16s}{s^2+4s+8}$$

Fig. 3-9. Algebraic Equations for Solution of F_T Matrix of Example 3.1

$$f_{14}s^2 + (f_{11} + 0.031 f_{14})s + (0.031 f_{11} + 0.26 f_{12}) = -1.969s^2 - 2.061s - 0.062$$

$$f_{15}s + f_{13} = -(s+1) \left[\frac{s^2 + 4s + 8}{s^2 + 4s + 4} \right]$$

$$f_{24}s^2 + (f_{21} + 0.031 f_{24})s + (0.031 f_{21} + 0.26 f_{22}) = 0$$

$$f_{25}s + f_{23} = -2s - 8$$

Fig. 3-10. Equations in Row Echelon Form for Solution of F_T Matrix of Example 3.1

$$F_T = \begin{bmatrix} -2 & 0 & -\frac{s^2+4s+8}{s^2+4s+4} & -1.969 & -\frac{s^2+4s+8}{s^2+4s+4} \\ 0 & 0 & -8 & 0 & -2 \end{bmatrix} \quad (3-78)$$

Also, from Eq. (3-51), the G_T matrix is:

$$G_T = \begin{bmatrix} 2 & 4 \\ 0 & 8 \end{bmatrix} \quad (3-79)$$

The application of the feedback and gain matrices, F_T and G_T above, to the original multivariable system will result in a closed-loop system corresponding to the desired transfer matrix of Eq. (3-74).

(End Example 3-1)

A simple solution for F_T results when the F_k matrices are equal for $k = 1, 2, \dots, m$. Then Eq. (3-55) can be written as

$$F_T \sum_{k=1}^m D_k = F \sum_{k=1}^m D_k \quad (3-80)$$

or,

$$F_T = F \quad (3-81)$$

where

$$F = F_1 = F_2 = \dots = F_m \quad (3-82)$$

A situation where the F_k matrices are equal for $k = 1, 2, \dots, m$ is encountered when the rows of the desired transfer matrix $H_d(s, F_T, G_T)$ are

identical except for the gain constants. Because of the significant computational simplification and because this type of cross-coupling can have many practical ramifications, the following theorem is presented.

Theorem 3.3

If all the elements within a given row of the desired transfer matrix $H_d(s, F_T, G_T)$ are identical except for their gain constants, the resultant set of F_k matrices are necessarily equal for $k = 1, 2, \dots, m$. The dynamic portion of the elements need not be identical from row to row.

Proof:

From Eq. (3-46), it is apparent that if the m F_k matrices are equal, then

$$B_k^{*-1} \hat{F}_k T_k = B_1^{*-1} \hat{F}_1 T_1 \quad (3-83)$$

for $k = 1, 2, \dots, m$. The development of the proof consists of constructing a desired transfer function as hypothesized, determining the relationship between the $\hat{F}_k$ and $\hat{F}_1$, T_k and T_1 matrices, and using Eq. (3-31) to form the product of Eq. (3-83).

Assume that the transfer functions h_{ij} of the general $H_d(s, F_T, G_T)$ transfer matrix given by Eq. (3-2) are represented as

$$h_{ij} = g_{ij} w_{ij}(s) \quad (3-84)$$

where g_{ij} represents the transfer function gain constant, and $w_{ij}(s)$ is a polynomial in s containing the desired poles and zeros of the $(ij)^{th}$

transfer function. The hypothesis of the theorem states that the $w_{ij}(s)$ for $j = 1, 2, \dots, m$ are equal although they need not be the same for each i , $i = 1, 2, \dots, m$. Therefore, using Eq. (3-84), $H_d(s, F_T, G_T)$ becomes

$$H_d(s, F_T, G_T) = \begin{bmatrix} g_{11}w_{11}(s) & g_{12}w_{11}(s) & \dots & g_{1m}w_{11}(s) \\ g_{21}w_{22}(s) & g_{22}w_{22}(s) & \dots & g_{2m}w_{22}(s) \\ \vdots & \vdots & & \vdots \\ g_{m1}w_{mm}(s) & g_{m2}w_{mm}(s) & \dots & g_{mm}w_{mm}(s) \end{bmatrix} \quad (3-85)$$

with $w_{ij}(s) = w_{ii}(s)$. Consequently, the $\bar{H}_k(s, F_k, G_k)$ transfer matrices of Eq. (3-10) are:

$$\bar{H}_1(s, F_1, G_1) = \begin{bmatrix} g_{11}w_{11}(s) & 0 & \dots\dots\dots & 0 \\ 0 & g_{22}w_{22}(s) & \dots\dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots\dots & g_{mm}w_{mm}(s) \end{bmatrix} \quad (3-86)$$

$$\bar{H}_2(s, F_2, G_2) = \begin{bmatrix} g_{21}w_{22}(s) & 0 & \dots\dots\dots & 0 & 0 \\ 0 & g_{32}w_{33}(s) & \dots\dots\dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots\dots\dots & g_{m,m-1}w_{mm}(s) & 0 \\ 0 & 0 & \dots\dots\dots & 0 & g_{1m}w_{11}(s) \end{bmatrix}$$

$$\vdots$$

$$\vdots$$

$$\bar{H}_m(s, F_m, G_m) = \begin{bmatrix} g_{m1}w_{mm}(s) & 0 & \dots\dots\dots & 0 \\ 0 & g_{12}w_{11}(s) & \dots\dots\dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots\dots\dots & g_{m-1,m}w_{m-1,m-1}(s) \end{bmatrix}$$

Now, $\hat{F}_{k_{ii}}$ is the feedback submatrix of $\hat{F}_k$ which positions the open-loop canonically decoupled subsystem poles in order to achieve the desired closed-loop transfer function of the i^{th} diagonal element in $\bar{H}_k(s, F_k, G_k)$. Let $\hat{F}_1$ be represented, using Fig. 2-5, as

$$\hat{F}_1 = \begin{bmatrix} \hat{F}_{111} & 0 & \dots & 0 & 0 \\ 0 & F_{122} & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & F_{1nm} & 0 \end{bmatrix} \quad (3-87)$$

where $\hat{F}_{1ii}$ is a $(p_{1i} \times p_{1i})$ submatrix, and each $\hat{F}_{1ii}$ produces $w_{ii}(s)$ for $i = 1, 2, \dots, m$. Then from Eq. (3-86), the structure of each $\hat{F}_k$ matrix can be determined in terms of the $\hat{F}_{1ii}$ submatrices as shown in Fig. 3-11.

Next, consider the structure of the transformation matrix T_1 , where using Eq. (2-51),

$$T_1 = \begin{bmatrix} T_{11} \\ T_{12} \\ \vdots \\ T_1 \\ T_{1m+1} \end{bmatrix} \quad (3-88)$$

The first $(d_{1i} + 1)$ rows of each of the $(p_{1i} \times n)$ submatrices T_{1i} in Eq. (3-88) are the rows $C_{1i}, C_{1i}\bar{A}, \dots, C_{1i}\bar{A}^{d_{1i}}$ where $\bar{A} = A + BF_1^*$. The remaining $(p_{1i} - d_{1i} - 1)$ rows are any linearly independent row vectors which make the set of rows of T_{1i} a basis for τ_{1i} .

Also, considering T_k , the first $(d_{ki} + 1)$ rows of each of the $(p_{ki} \times n)$ submatrices T_{ki} , $k = 2, 3, \dots, m$, are the rows $C_{ki}, C_{ki}\bar{A}, \dots, C_{ki}\bar{A}^{d_{ki}}$, where $\bar{A}$ again equals $A + BF_1^*$. From Eq. (3-28),

$$\hat{F}_2 = \begin{bmatrix} \hat{F}_{122} & 0 & \cdot & \cdot & \cdot & 0 & 0 & 0 \\ 0 & \hat{F}_{133} & \cdot & \cdot & \cdot & 0 & 0 & 0 \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & \cdot & \hat{F}_{1mm} & 0 & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 & \hat{F}_{111} & 0 \end{bmatrix}$$

$$\hat{F}_3 = \begin{bmatrix} \hat{F}_{133} & 0 & \cdot & \cdot & \cdot & 0 & 0 & 0 & 0 \\ 0 & \hat{F}_{144} & \cdot & \cdot & \cdot & 0 & 0 & 0 & 0 \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & \cdot & \hat{F}_{1mm} & 0 & 0 & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 & \hat{F}_{111} & 0 & 0 \\ 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 & \hat{F}_{122} & 0 \end{bmatrix}$$

$$\begin{matrix} \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{matrix}$$

$$\hat{F}_m = \begin{bmatrix} \hat{F}_{1mm} & 0 & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & \hat{F}_{111} & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & & & & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot \\ \cdot & \cdot & & & & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & \cdot & \hat{F}_{1m-1,m-1} & 0 \end{bmatrix}$$

Fig. 3-11. $\hat{F}_k$ Matrices for Theorem 3.3

C_{k_i} is equivalent to C_{1_i} where $j = i+k-1$ for $i \leq m-k+1$ or $j = i+k-m-1$ for $i \geq m-k+1$. Also, if the same set of $(p_{1_j} - d_{1_j} - 1)$ independent row vectors used to make the set of rows of T_{1_j} a basis for τ_{1_j} are used to make the set of rows of T_{k_i} a basis for τ_{k_i} , where again $j = i+k-1$ when $i \leq m-k+1$ or $j = i+k-m-1$ for $i \geq m-k+1$, the resultant T_k transformation matrices, $k = 2, 3, \dots, m$ are:

$$T_k = [T_{1_k}^T, T_{1_{k+1}}^T, \dots, T_{1_m}^T, T_{1_1}^T, \dots, T_{1_{k-1}}^T, T_{1_{m+1}}^T]^T \quad (3-89)$$

where each T_{1_i} is a $(p_{1_i} \times n)$ submatrix of T_1 . Therefore, the results for the products $\hat{F}_k T_k$, $k = 1, 2, \dots, m$, using Eqs. (3-87) and (3-89) and Fig. 3-11 are computed and given in Fig. 3-12. Using Eq. (3-31), where Q^{-1} is

$$Q^{-1} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix} \quad (3-90)$$

the results for the products $B_k^{*-1} \hat{F}_k T_k$ for $k = 2, 3, \dots, m$ are:

$$B_k^{*-1} \hat{F}_k T_k = B_1^{*-1} [(\hat{F}_{1_{11}} T_{1_1})^T, (\hat{F}_{1_{22}} T_{1_2})^T, \dots, (\hat{F}_{1_{mm}} T_{1_m})^T]^T \quad (3-91)$$

However, since the right-hand portion of Eq. (3-91) is $B_1^{*-1} \hat{F}_1 T_1$, the resulting relationship of Eq. (3-83) is obtained. Consequently, the result that $F_k = F_1$ for all $k = 1, 2, \dots, m$ is obtained, and the theorem is proven. (End Theorem 3.3)

$$\hat{F}_1 T_1 = [(\hat{F}_{1_{11}} T_{1_1})^T, (\hat{F}_{1_{22}} T_{1_2})^T, \dots, (\hat{F}_{1_{mm}} T_{1_m})^T, 0]^T$$

$$\hat{F}_2 T_2 = [(\hat{F}_{1_{22}} T_{1_2})^T, (\hat{F}_{1_{33}} T_{1_3})^T, \dots, (\hat{F}_{1_{mm}} T_{1_m})^T, (\hat{F}_{1_{11}} T_{1_1})^T, 0]^T$$

$$\vdots \quad \quad \quad \vdots$$

$$\hat{F}_k T_k = [(\hat{F}_{1_{kk}} T_{1_k})^T, \dots, (\hat{F}_{1_{mm}} T_{1_m})^T, (\hat{F}_{1_{11}} T_{1_1})^T, \dots, (\hat{F}_{1_{k-1,k-1}} T_{1_{k-1}})^T, 0]^T$$

$$\vdots \quad \quad \quad \vdots$$

$$\hat{F}_m T_m = [(\hat{F}_{1_{mm}} T_{1_m})^T, (\hat{F}_{1_{11}} T_{1_1})^T, \dots, (\hat{F}_{1_{m-1,m-1}} T_{1_{m-1}})^T, 0]^T$$

where

$\hat{F}_{1_{ii}} T_{1_i}$ is a $(1 \times n)$ submatrix

Fig. 3-12. $\hat{F}_k T_k$ Matrices for $k = 1, 2, \dots, m$ of Theorem 3.3

Example 3.2

To illustrate an example where the F_k matrices are equal for $k = 1, 2, \dots, m$, consider again the multivariable system of Example 3.1. This time, however, assume that the desired transfer matrix to be realized by F_T and G_T is

$$H_d(s, F_T, G_T) = \begin{bmatrix} \frac{2}{s^2 + 2s + 2} & \frac{4}{s^2 + 2s + 2} \\ \frac{3}{s^2 + 4s + 8} & \frac{8}{s^2 + 4s + 8} \end{bmatrix} \quad (3-92)$$

where zero steady-state error is not considered a necessary design requirement for the off-diagonal transfer functions. Repeating the variational decoupling technique, the resultant F_k and G_k matrices are:

$$F_1 = \begin{bmatrix} -2 & 0 & 0 & -1.969 & 0 \\ 0 & 0 & -8 & 0 & -2 \end{bmatrix} = F_2 \quad (3-93)$$

and

$$G_1 = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}, \quad G_2 = \begin{bmatrix} 0 & 4 \\ 3 & 0 \end{bmatrix} \quad (3-94)$$

The F_T and G_T matrices are consequently simply:

$$F_T = \begin{bmatrix} -2 & 0 & 0 & -1.969 & 0 \\ 0 & 0 & -8 & 0 & -2 \end{bmatrix} \quad (3-95)$$

and

$$G_T = \begin{bmatrix} 2 & 4 \\ 3 & 8 \end{bmatrix} \quad (3-96)$$

These results illustrate that when the desired transfer matrix possesses identical row dynamics, no dynamic elements are required in the feedback matrix F_T . This result is general and a consequence of Theorem 3.3.

(End Example 3.2)

A closed-form solution for F_T will exist under the restriction that $m = n$ for the multivariable system. Then the matrix $\sum_{k=1}^m D_k$ is square, and its inverse can be determined provided that the $\sum_{k=1}^m D_k$ matrix is nonsingular. Then

$$F_T = \left[\sum_{k=1}^m F_k D_k \right] \left[\sum_{k=1}^m D_k \right]^{-1} \quad (3-97)$$

The computation of F_T using Eq. (3-97) must be considered a very special case since the number of inputs or outputs does not usually equal the number of system state variables for a physically real multivariable system. Also, the type of transfer matrices obtainable are not usually desirable since only one pole can exist at most in each subsystem. However, if each pole is positioned as desired, additional open-loop dynamic compensators can be placed in the input channels to condition the system response further (Tyler and Tuteur, 1966).

Summary

The general design equations necessary to obtain a closed-loop multivariable control-system possessing input-output cross-coupling are formulated. The formulation is based on the decoupling techniques presented in Chapter 2. Theorems are presented to establish the validity of the design technique. The results obtained in this chapter represent new contributions to the area of state-variable feedback design of multivariable control-systems and are the main contributions of this study.

The variational decoupling technique provides a logical mathematical procedure for decoupling a given multivariable system into all possible relationships between the system outputs and the control inputs. A specific elementary row transformation matrix is introduced to generate a restricted set of $(m-1)$ partial vector-matrix representations from a reference vector-matrix system representation. The set is restricted in the sense that if the m decoupling matrices, F_k and G_k , $k = 1, 2, \dots, m$, are computed, every element in the input vector will be decoupled with a different element in the output vector without repetition over the entire range of k . A step-by-step procedure is indicated for calculating the set of F_k and G_k matrices.

Equations are derived for the control-law matrices, F_T and G_T , which result in a closed-loop transfer matrix containing the desired transfer functions between all system outputs and external control inputs. The equations relate F_T and G_T to the known m pairs of F_k and G_k matrices resulting from the variational decoupling technique. The

formulation indicates that in general, a closed form solution of F_T does not exist. Consequently, a computational method for computing a satisfactory F_T matrix is discussed. The method consists of applying the Gaussian elimination process to the algebraic equations resulting from the formulation for the F_T matrix. After the equations are in row echelon form, the coefficients of like powers of s are equated and those equations are then solved for the elements of F_T . The design method is illustrated using the set of differential equations for the example of Chapter 2.

The equation for F_T in terms of the F_k and G_k matrices requires the computation of m state transfer matrices D_k , where $D_k = [sI_n - A - BF_k]^{-1} BG_k$. A brief discussion is presented on the computation of the D_k matrices since the computation requires taking an inverse of a large matrix containing elements that are a function of the Laplace variable s . The computation is shown to be reduced in complexity if the set of D_k matrices are computed in terms of the canonical system matrices $\hat{A}_k$, $\hat{B}_k$, and $\hat{C}_k$ and the transformation matrix T_k .

Finally, two special cases are presented which result in a closed form solution for F_T . The most important of these is the situation where the elements within each row of the desired transfer matrix have identical row dynamics. Every row can have different dynamics, but if each element in any row of the desired transfer matrix differs only in the gain constant, F_T will be equal to F_k , for all m values of k .

CHAPTER 4

THERMIONIC REACTOR CONTROL SYSTEM

The purpose of this chapter is to apply the design technique for an interacting multivariable control system presented in the previous chapter on a practical example. The example consists of a somewhat reduced version of the thermionic reactor model utilized in the start-up analysis performed by Guppy (1970) and in the single-input, single-output control system study by Weaver, Gronroos, Guppy, and Davis (1969).

In particular, the mathematical model consists of an in-core, single-diode thermionic reactor representation with the reactor kinetics described by a point reactor model using the "prompt-jump" approximation (Hetrick, 1971). The control inputs are reactivity, diode load resistance, and coolant temperature. The system outputs, assumed to be measurable either directly or indirectly, are neutron power density, diode emitter temperature, and diode collector temperature.

The design of the control system depends upon the control philosophy used to adjust for load changes, coolant loss, diode failure, power excursions, etc. In this study, the physical constraints used to form the desired transfer matrix consisted of 1) a constant diode voltage, 2) a thermal shock limitation of less than 60°C/minute in the diode collector and emitter temperature, and 3) a separation of the thermal power and diode electrical power dependency for sudden

load changes. The set of independent input perturbations consisted of 1) a 2¢/second ramp of reactivity for a duration of 20 seconds, 2) a 20% step reduction in the diode load resistance, and 3) a 1°C/second ramp change in the coolant temperature for a duration of 20 seconds. The desired transfer matrix was chosen such that the resultant feedback and gain matrices consisted of constant elements.

The approach to the control system design was to linearize the equations describing the thermionic reactor system and to then compute the state feedback and input gain matrices resulting in a control-law which satisfied the desired constraints. The results are presented graphically and consist of linear time responses for the given input perturbations.

Diode Representation

A simple schematic taken from Guppy (1970) which illustrates the various temperatures and regions of the model is shown in Fig. 4-1. The model used in this control study is reduced from that used in the start-up analysis in the following ways: 1) the coolant and radiator state equations are excluded, and 2) the insulator and cladding regions are collapsed and treated as part of the diode collector region. The basis for the first reduction is that the coolant loop is generally sluggish and consequently does not present a difficult control problem. The inclusion of the coolant loop equations in the model would represent primarily an added mathematical complexity which is not warranted for this initial study. The second simplification is used because of the inherent coupling between the collector surface temperature and the

Fig. 4.1. Representation of Regions for Single Diode Approximation

bulk coolant temperature. Therefore, any change in coolant temperature will induce a similar change in the collector temperature and vice versa with no external control present. It is noted here that the simplification is not needed since the decoupling techniques are still applicable without it. The decoupling techniques indicate however that the collector, insulator, and cladding states are inseparable and that they always occur in the same subsystem. It is felt that the resultant reduction in the order of the model would simplify the required computations in the control study and consequently the assumption is made.

The remaining state-equations used to describe the fuel-diode region are taken from the above-cited reference by Guppy, and they represent the starting point of this control study. If the reader is interested in the details of the system model development, he is referred to the reference by Guppy (1970) or by Brehm, et al. (1970). The values of the material parameters are assumed to be temperature independent and represent values typically associated with a uranium-carbide fuel region, a tungsten emitter, a niobium collector, and a cesium reservoir gas at nominal operating conditions. The state-equations used in this study are:

$$\dot{C} = n - 0.1 c \quad (4-1)$$

$$\dot{\bar{T}}_F = 0.377 n - 0.303 (\bar{T}_F - \bar{T}_E) \quad (4-2)$$

$$\dot{\bar{T}}_E = 1.722 (\bar{T}_F - \bar{T}_E) - 3.05 q_E'' \quad (4-3)$$

$$\dot{\bar{T}}_c = 3.714 (q_E'' - IV) - 4.773 (\bar{T}_c - \bar{T}_{co}) \quad (4-4)$$

where

$$C = \frac{\lambda C'}{\beta} \quad (4-5)$$

and

C' = average neutron precursor density in fuel region,
watts/cm³

n = average neutron power density, watts/cm³

$\bar{T}_F$ = average reactor fuel region temperature, °K

$\bar{T}_E$ = average diode emitter region temperature, °K

$\bar{T}_c$ = average diode collector region temperature, °K

$\bar{T}_{co}$ = axial averaged bulk coolant temperature, °K

q_E'' = diode emitter heat flux, watts/cm²

IV = diode electrical power, watts/cm²

λ = neutron generation time, $\sim 10^{-7}$ secs

β = delayed neutron fraction, 0.0064

The equilibrium values for the state-equation variables, C_0 , n_0 ,

$\bar{T}_{F_0}$, $\bar{T}_{E_0}$, $\bar{T}_{c_0}$, $\bar{T}_{co_0}$, q_{E_0}'' , and $I_0 V_0$, are defined by setting all the derivative quantities in Eqs. (4-1) to (4-4) equal to zero.

Since the neutron generation time is very small for this system, a relationship for the neutron power in terms of the precursor concentration and system reactivity is obtained by applying the "prompt-jump" approximation to the point reactor equations (Hetrick, 1971) used by Guppy, where:

$$n = \frac{\lambda C}{1 - \rho(t)} \quad (4-6)$$

with

$\rho(t)$ = reactivity of the reactor system in fractions of β from equilibrium, ($\$$)

λ = effective delayed neutron precursor decay constant, 0.1 sec^{-1}

Also, the system reactivity, ρ , is expressed as

$$\rho = \alpha_F(\bar{T}_F - \bar{T}_{F_0}) + \alpha_E(\bar{T}_E - \bar{T}_{E_0}) + \alpha_C(\bar{T}_C - \bar{T}_{C_0}) + \rho_{\text{con}} \quad (4-7)$$

where

α_F = prompt temperature-reactivity feedback coefficient,
 $+ 4. \times 10^{-5} \$/^{\circ}\text{K}$

α_E = emitter region temperature-reactivity feedback coefficient,
 $- 8. \times 10^{-5} \$/^{\circ}\text{K}$

α_C = collector region temperature-reactivity feedback
 coefficient, $-1.6 \times 10^{-4} \$/^{\circ}\text{K}$

ρ_{con} = external reactivity contribution, $\$$

In order for the system to be at equilibrium, ρ must be zero; hence at equilibrium, ρ_{con} is zero. The set of nonlinear equations given above represents the in-core thermionic reactor system used in this control study.

The computation of the quantities q_E'' and IV is accomplished with a digital computer code called SIMCON, written by Wilkins (1968). The code has been extensively tested against experimental results and has proven successful in simulating performance characteristics for the types of diodes considered in this study over a wide range of operating conditions. Basically, the computer code computes the values of q_E'' and IV for different values of the diode current, I , given the five

input parameters, $\bar{T}_E$, $\bar{T}_C$, $\bar{T}_R$, d , and R_L . The new symbols, $\bar{T}_R$, d , and R_L , are the cesium reservoir temperature, the diode gap distance, and diode load resistance, respectively.

In order to apply the design techniques for an interacting multivariable control system, the system equations must be linearized and placed in a form compatible with Eqs. (2-1) and (2-2). The methodology of this linearization consists of taking the system to be initially at equilibrium and then assuming that a small input perturbation is applied to the system such that the time response trajectory of the system states remains in the neighborhood of the initial equilibrium point. The neighborhood of the equilibrium point is defined as the region in state-space around the equilibrium point in which the effect of the nonlinearities is small. Consequently, the nonlinear system representation will agree approximately with a linearized version of the same system for small perturbations.

Before proceeding with a mathematical development of the linearization of the system, it is imperative to define numerically the position of the equilibrium point in the state-space. The equilibrium point is obtained by choosing the equilibrium temperatures, $\bar{T}_{E_0}$, $\bar{T}_{C_0}$, and $\bar{T}_{R_0}$, to be $\bar{T}_{E_0} = 2000^\circ\text{K}$, $\bar{T}_{C_0} = 1000^\circ\text{K}$, and $\bar{T}_{R_0} = 630^\circ\text{K}$. Also, the gap distance d is fixed at 10 mils, and the diode load resistance R_{L_0} is specified at 0.035 ohm-cm^2 . With these equilibrium values, the SIMCON program was used to compute the equilibrium values, I_0 , V_0 , and q_{E_0}'' . The values for C_0 , n_0 , $\bar{T}_{F_0}$, and $\bar{T}_{co_0}$ are obtained by solving Eqs. (4-1), (4-2), (4-4), and (4-6), with the derivative terms set to zero. The results are tabulated below in Table 4-1.

Table 4-1. Initial Equilibrium Conditions for System

n_0 (watts/cm ³)	C_0 ($\frac{\text{watt-sec}}{\text{cm}^3}$)	$\bar{T}_{F0}$ (°K)	$\bar{T}_{E0}$ (°K)	$\bar{T}_{c0}$ (°K)
83.54	835.4	2104	2000	1000
$\bar{T}_{co0}$ (°K)	$\bar{T}_{R0}$ (°K)	I_0 (amps/cm ²)	V_0 (volts)	q_{E0}'' (watts/cm ²)
891.5	630	19.3	0.6775	67.667

The value of the cesium reservoir temperature $\bar{T}_{R0}$ represents a higher-than optimum value and ensures that the diode would remain in the ignited mode for a $\pm 5^\circ\text{K}$ variation in $\bar{T}_R$.

The following definitions are used to describe the deviation of the system states from the equilibrium point due to a small input perturbation. They are:

$$\begin{aligned}
 \delta n(t) &= n - n_0 & (4-8) \\
 x_1(t) &= C - C_0 \\
 x_2(t) &= \bar{T}_F - \bar{T}_{F0} \\
 x_3(t) &= \bar{T}_E - \bar{T}_{E0} \\
 x_4(t) &= \bar{T}_c - \bar{T}_{c0} \\
 u_2(t) &= R_L - R_{L0} \\
 u_3(t) &= \bar{T}_{co} - \bar{T}_{co0}
 \end{aligned}$$

$$\delta q_E'' = q_E'' - q_{E_0}''$$

$$\delta(\text{IV}) = \text{IV} - \text{I}_0 V_0$$

$$\delta\rho = \rho(t)$$

The linearization of Eq. (4-6) is accomplished using the definitions of $\delta n(t)$, $\delta\rho$, and $x_1(t)$ from Eq. (4-8). Substituting for n , C , and $\rho(t)$ in Eq. (4-6) yields:

$$n_0 + \delta n(t) - n_0 \delta\rho - \delta\rho \delta n(t) = 0.1 C_0 + 0.1 x_1(t) \quad (4-9)$$

But from Eq. (4-1), at equilibrium n_0 equals $0.1 C_0$, and neglecting the product $\delta\rho \delta n(t)$, Eq. (4-9) becomes:

$$\delta n(t) = 0.1 x_1(t) + n_0 \delta\rho \quad (4-10)$$

Substituting Eq. (4-10) into Eqs. (4-1) and (4-2), then using again the definitions of Eq. (4-8), results in:

$$\dot{x}_1(t) = n_0 \delta\rho \quad (4-11)$$

$$\dot{x}_2(t) = 0.0377 x_1(t) + 0.377 n_0 \delta\rho - 0.303 (x_2(t) - x_3(t)) \quad (4-12)$$

However, from Eq. (4-7), using again Eq. (4-8):

$$\delta\rho = \alpha_F x_2(t) + \alpha_E x_3(t) + \alpha_C x_4(t) + u_1(t) \quad (4-13)$$

where $u_1(t)$ is defined as the external reactivity contribution, ρ_{con} . Substituting Eq. (4-13) into Eqs. (4-11) and (4-12), with the numerical values for n_0 , α_F , α_E , α_C , results in the linearized form of Eqs. (4-1) and (4-2):

$$\begin{aligned} \dot{x}_1(t) = & 0.003342 x_2(t) - 0.006683 x_3(t) - 0.01337 x_4(t) \\ & + 83.54 u_1(t) \end{aligned} \quad (4-14)$$

$$\begin{aligned}\dot{x}_2(t) = & 0.0377 x_1(t) - 0.3017 x_2(t) + 0.3005 x_3(t) \\ & - 0.00504 x_4(t) + 31.49 u_1(t)\end{aligned}\quad (4-15)$$

The linearized expressions for $\delta q_E''$ and $\delta(IV)$ are obtained by using a first-order Taylor expansion, neglecting cross-product terms, and then computing the coefficients with the SIMCON computer code (Wilkins, 1968). In the SIMCON code, the diode heat flux or the electron cooling term, $\delta q_E''$, and the diode electrical power $\delta(IV)$, are dependent on the diode emitter, diode collector, and cesium reservoir temperatures and on the diode load resistance. Since the cesium reservoir temperature is assumed to remain constant during all system transients, the resultant expansions are:

$$\delta q_E'' = \frac{\Delta q_E''}{\Delta T_E} x_3(t) + \frac{\Delta q_E''}{\Delta T_C} x_4(t) + \frac{\Delta q_E''}{\Delta R_L} u_2(t) \quad (4-16)$$

and

$$\delta(IV) = \frac{\Delta(IV)}{\Delta T_E} x_3(t) + \frac{\Delta(IV)}{\Delta T_C} x_4(t) + \frac{\Delta(IV)}{\Delta R_L} u_2(t) \quad (4-17)$$

Results from the SIMCON code used to compute the coefficients are given in Table 4-2. The ΔT_E , ΔT_C , and ΔR_L variations used to compute the coefficients are 50°K, 100°K, and 0.01 ohms/cm², respectively. Substituting the numerical values for the coefficients in Eqs. (4-16) and (4-17) results in:

$$\delta q_E'' = 0.1766 x_3(t) + 0.02443 x_4(t) - 1263 u_2(t) \quad (4-18)$$

$$\delta(IV) = 0.0526 x_3(t) + 0.01169 x_4(t) - 170 u_2(t) \quad (4-19)$$

Table 4-2. SIMCON Results for Taylor Expansion Coefficients

$T_E, ^\circ\text{K}$	$T_c, ^\circ\text{K}$	$R_L, \Omega/\text{cm}^2$	$q_E'', \text{ watts/cm}^2$	IV, watts
2000	1000	0.035	67.667	13.023
2025	1000	0.035	72.040	14.360
1975	1000	0.035	63.210	11.730
2000	1050	0.035	68.651	13.510
2000	950	0.035	66.208	12.341
2000	1000	0.040	62.724	12.280
2000	1000	0.030	73.883	13.907

The linearized form of Eqs. (4-3) and (4-4) are obtained by using the definitions of Eq. (4-8) and the expressions for $\delta q_E''$ and $\delta(\text{IV})$ where

$$\dot{x}_3(t) = 1.722 x_2(t) - 2.261 x_3(t) - 0.0745 x_4(t) + 3852 u_2(t) \quad (4-20)$$

$$\dot{x}_4(t) = 0.4605 x_3(t) - 4.726 x_4(t) - 4059 u_2(t) + 4.773 u_3(t) \quad (4-21)$$

The state variable system output relationships in terms of the defined quantities in Eq. (4-8) are:

$$y_1(t) = x_1(t) \quad (4-22)$$

$$y_2(t) = x_3(t)$$

$$y_3(t) = x_4(t)$$

Consequently, the state-equations of Eqs. (4-14), (4-15), (4-20), and (4-21) and the system output relationships of Eq. (4-22) constitute the linear representation of the reduced thermionic reactor

system. Before proceeding with the actual control system computations, possible control-system objectives are examined to aid in the formulation of the desired closed-loop system transfer matrix.

Control System Objectives

In this section some typical plausible objectives of the desired control system are discussed in order to provide a basis for choosing the desired closed-loop transfer matrix used in this study. It should be emphasized that the primary purpose of this chapter is to illustrate the applicability of the interacting design technique of Chapter 3 on a meaningful example and not to present an actual control-system design for an in-core thermionic reactor system. The development presented below is based on the premise that the system model is sufficiently accurate to warrant following general procedures which would be used to formulate the control system design objectives and the desired transfer matrix in an actual control-system design.

The control objectives are specified in part by considering a potential application of the thermionic reactor system, the range and magnitudes of possible control input signals, and the inherent physical and thermal constraints of the system materials. Cost constraints and sensitivity analysis of parameter variations are not used in deciding the control philosophy of this study.

Assume that the thermionic reactor power plant is to be used to provide electrical power for space ion-engine applications. A constant voltage electrical power source may be desirable to regulate the thrust of the ion-engine system and to provide a constant power source for

experimental equipment, instrumentation, etc. The greater share of the reactor electrical power is to be used in the propulsion system, with possibly a large portion of the mission time being spent in a coast mode. But the thermionic diode current-voltage characteristics indicate that the diode voltage and efficiency for a given set of operating temperatures vary considerably for resistive load changes. Therefore, an important control objective would be to minimize the effect of load changes due to the start-up and shut-down of ion engines.

Other control objectives arise by considering the physical and thermal constraints inherent in the fuel-diode regions indicated in Fig. 4-1. Suppose that the fuel region is composed of sintered uranium carbide with a melting point of 2650°K . A control objective might be to limit the centerline temperature to approximately 2450°K to ensure that no fuel melting will occur. Although the fuel temperature is not a controlled variable in this study, this constraint can be satisfied if the control system forces the state of neutron power density to be less than some maximum value above which fuel melting might occur.

The control system might also be used to keep the emitter temperature to within some specified deviation from the equilibrium temperature in order to prevent thermal cycling of the diode and fuel to preserve the validity of the theory upon which the control system design is based.

A control system which could limit system thermal shocks to some prespecified tolerable level, particularly in the emitter, collector, and liquid metal coolant loop components, would be highly desirable.

The list of control objectives would be much longer in an actual control system design, but a sufficient number have been specified for this study to proceed with the specification of the desired transfer matrix.

The selection of the desired transfer matrix is formulated on the hypothesis that the closed-loop control system is subjected to three types of external input signals. The input signals represent 1) a 2¢/second ramp of reactivity for a duration of 20 seconds, 2) a 20% step reduction in the diode load resistance, and 3) a 1°C/second ramp-increase in the coolant temperature for a duration of 20 seconds. The system is initially assumed to be at the equilibrium point in state-space given in Table 4-1, and each input signal is applied separately. The step reduction in load resistance is a perturbation input and is used to determine how well the control system can adjust for load changes possibly resulting from changing thrust requirements of the ion-engines. Also, the rise in coolant temperature can be considered a perturbation input arising from a sudden reduction of coolant pumping power. The reactivity input signal represents a controlling signal to the reactor control device to initiate or control thermal power changes.

Desired Transfer Matrix Formulation

The desired transfer matrix is obtained by choosing the transfer functions relating each system output to all the control inputs. For the system model developed above, this means picking nine transfer functions, since the system has three open-loop control inputs and three outputs. Zero steady-state error is not considered important

in this control-system design, and therefore the transfer matrix is formulated to have identical row dynamics in order that the feedback matrix, F_T , will contain no time-dependent elements (see Example 3.2). Also, in order to use design specifications such as steady-state output values, natural frequency, damping ratio, etc., all the transfer functions are specified to be second-order responses.

The steady-state values of the system outputs $y_1(t)$, $y_2(t)$, and $y_3(t)$ are denoted $y_{1\text{ss}}(\infty)$, $y_{2\text{ss}}(\infty)$, and $y_{3\text{ss}}(\infty)$, respectively. The $y_{i\text{ss}}(\infty)$ values for $i = 1, 2, 3$ are fixed so that they are less than the deviations used to compute the coefficients of the first-order Taylor expansion approximation for q_E'' and IV (Table 4-1). Also, by restricting the emitter and collector temperature variations to the neighborhood of the equilibrium point for the set of input signals, it is hypothesized that the diode voltage will not change by more than 4 or 5%, and consequently the constraint of constant diode voltage will be approximately satisfied. The validity of this will be demonstrated with the time response curves.

The values for the damping ratio, δ , are chosen so that the thermal shock constraints are satisfied by using damping ratios that are greater than unity ($\delta > 1$), and consequently the time responses of the controlled variables are over-damped.

The last parameter necessary to compute the desired transfer matrix is the undamped natural frequency, ω_n , which represents the frequency of oscillation when the damping ratio is zero. The values of

ω_n are chosen so that the system gains used to obtain the steady-state output values are not excessively large.

The numerical values for δ , ω_n , and $y_{i_{ss}}(\infty)$ for each system output and control input combination are compiled in Table 4-3 below:

Table 4-3. Design Parameters for Desired Transfer Matrix

	$y_1(t)$	$y_2(t)$	$y_3(t)$	
$r_1(t)$	0.8	0.5	1.0	ω_n
	2.5	2.6	2.8	δ
	10.0	10.0	-20.0	$y_{i_{ss}}(\infty)$
$r_2(t)$	0.8	0.5	1.0	ω_n
	2.5	2.6	2.8	δ
	0	10.0	10.0	$y_{i_{ss}}(\infty)$
$r_3(t)$	0.8	0.5	1.0	ω_n
	2.5	2.6	2.8	δ
	10.0	0	10.0	$y_{i_{ss}}(\infty)$

In order to separate the direct open-loop coupling between the reactor and the thermionic diode complex, the power density variation reflected through the precursor concentration density $y_1(t)$ is chosen to be zero for the electric load perturbation $r_2(t)$. Also, a zero response is chosen at the system output $y_2(t)$ corresponding to the emitter temperature state when the positive coolant temperature ramp $r_3(t)$ is applied to the system.

The transfer function between the i^{th} system output $y_i(t)$ and the j^{th} external input r_j for $i = 1, 2, 3$ and $j = 1, 2, 3$ is computed using Eq. (4-23) below:

$$y_i(s)/r_j(s) = \frac{g_{ij}}{s^2 + 2\delta_{ij}\omega_{n_{ij}}s + \omega_{n_{ij}}^2} \quad (4-23)$$

where the subscripts i and j on δ and ω_n denote the value in Table 4-3 corresponding to the i^{th} system output and the j^{th} control input. Also, expressing the final value for the external control input $r_j(t)$ as $r_{j_f}(\infty)$, $j = 1, 2, 3$, the gain constants g_{ij} are computed using Eq. (4-24) below:

$$g_{ij} = \omega_{n_{ij}}^2 y_{i_{ss}}(\infty)/r_{j_f}(\infty) \quad (4-24)$$

The values of $r_{j_f}(\infty)$ for $j = 1, 2, 3$ are \$.40, $-0.007 \Omega\text{-cm}^2$, and 20°K , respectively.

For the values of $r_{j_f}(\infty)$ and the values in Table 4-3, the desired transfer matrix becomes:

$$y_d(s) = \begin{bmatrix} \frac{160}{s^2 + 4s + 0.64} & 0 & \frac{3.2}{s^2 + 4s + 0.64} \\ \frac{6.25}{s^2 + 2.6s + 0.25} & \frac{-357.143}{s^2 + 2.6s + 0.25} & 0 \\ \frac{-50}{s^2 + 5.6s + 1} & \frac{-1428.57}{s^2 + 5.6s + 1} & \frac{0.5}{s^2 + 5.6s + 1} \end{bmatrix} r(s) \quad (4-25)$$

The control-system design now consists of computing the F_T and G_T matrices for the set of equations given by Eqs. (4-14), (4-15), (4-20), and (4-21) such that the closed-loop transfer matrix is identical to the transfer matrix of Eq. (4-25).

Calculation of Control-Law Matrices

This subsection presents the computational results for the feedback and gain matrices F_T and G_T , required to realize the closed-loop transfer matrix of Eq. (4-25). The computation is not developed in detail because previous examples in Chapter 3 have been presented to indicate the procedure. Consequently, the resultant F_T and G_T matrices are given without including the intermediate steps in the calculation. The validity of the F_T and G_T matrices is established by computing the linear time responses for the set of input control signals and observing that the responses agree with the transfer functions indicated in Eq. (4-25).

However, two general comments concerning the calculation of F_T and G_T are in order. First, three states of added dynamic compensation, each representing the addition of one subsystem pole, must be inserted into the control input channels of the integrator decoupled system. The dynamic compensation is necessary in order to achieve the second-order transfer functions of Eq. (4-25). Also, since the row dynamics are identical in the desired transfer matrix, Theorem 3.3 proves that the resultant F_k matrices are equal for $k=1,2,3$. The F_k matrices are the feedback matrices which result in the diagonal transfer matrices $\bar{H}_k(s, F_k, G_k)$ of

Eq. (3-10), where the $\bar{H}_k(s, F_k, G_k)$ matrices, $k = 1, 2, 3$, are obtained from the desired transfer matrix of Eq. (4-25). Consequently, the D_k matrices for $k = 1, 2, 3$, given by Eq. (3-54) are not computed. Instead, F_T is obtained from the F_k matrices as indicated in Eq. (3-81).

The resultant F_T and G_T matrices are:

$$F_T = \begin{bmatrix} -0.64 & 0 & 0 & 0 & -3 & 0 & 0 \\ 0 & 0 & -0.25 & 0 & 0 & -1.6 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -4.6 \end{bmatrix} \quad (4-26)$$

and

$$G_T = \begin{bmatrix} 160 & 0 & 3.2 \\ 6.25 & -357. & 0 \\ -50. & -1429. & 0.5 \end{bmatrix} \quad (4-27)$$

The next step is to apply the control-law of Eq. (3-13) with the F_T and G_T in Eqs. (4-26) and (4-27) to the differential equations of Eqs. (4-14), (4-15), (4-20), and (4-21), to determine the time responses for the set of inputs given in Section 1 of this chapter.

Closed-Loop System Time Responses

This section presents the linear time responses for the closed-loop control system state variables of neutron power density, average fuel temperature, and emitter and collector surface temperatures. The time responses result from the application of the control input signals discussed above.

The time responses are obtained by using a digital computer program to integrate the set of linear differential equations representing the thermionic reactor system. The integration routine consists of a very fast predictor-corrector iteration scheme capable of handling both linear and nonlinear first order differential equations. The program was written by Krogh (1969) at Jet Propulsion Laboratory, and a listing appears in the appendix of the study by Guppy (1970).

Ramp Change in Reactivity

The time responses due to the input signal $r_1(t)$ representing a $+2\text{¢/second}$ ramp of reactivity insertion for a time period of 20 seconds are presented in Fig. 4-2. The linear time responses of the controlled states, $x_1(t)$, $x_3(t)$, and $x_4(t)$ representing equilibrium changes in the precursor concentration and the emitter and collector temperatures, correspond to the desired transfer functions in the first column of the first column of the transfer matrix in Eq. (4-25). Although the precursor density variation $x_1(t)$ is not plotted, the resultant linear time response can be verified indirectly by using the plotted $\delta n(t)$ and $\delta \rho(t)$ values in Fig. 4-2 and computing $x_1(t)$ from Eq. (4-6) with the definitions of Eq. (4-8).

The time responses for the net system reactivity ($\delta \rho(t)$), the net diode load resistance ($R_L(t)$), and the bulk coolant temperature variation ($u_3(t)$) also appear in Fig. 4-2. The net system reactivity is related to the open-loop control signal $u_1(t)$ by Eq. (4-13). The relationship between $R_L(t)$ and $u_2(t)$ is indicated in Fig. 4-2. The

Fig. 4-2. Linear Responses for +2¢/second Reactivity Ramp of 20-Second Duration

Fig. 4-2. Continued

t, sec	I, amps/cm ²	V, volts	IV, watts/cm ²
0.0	19.30	0.676	13.04
5.0	19.61	0.672	13.28
10.0	20.18	0.667	13.45
15.0	20.78	0.661	13.73
20.0	21.39	0.654	14.01
25.0	21.65	0.653	14.15
30.0	21.62	0.655	14.17
35.0	21.59	0.656	14.17
40.0	21.58	0.657	14.18
45.0	21.58	0.658	14.19
50.0	21.58	0.658	14.20
55.0	21.58	0.658	14.20
60.0	21.58	0.658	14.20

Fig. 4-2. Continued

variations in $u_1(t)$, $u_2(t)$, and $u_3(t)$ represent the open-loop control signals, generated by the feedback and gain compensation matrices, F_T and G_T , required to produce the desired coupled, closed-loop responses at the controlled system states.

These results indicate the basic difference between a single-input, single-output and a multiple-input, multiple-output control system. For in this study, when one external input signal is applied to the system, three different open-loop control signals are generated, each of which interacts to produce a set of three independent desired system output responses. In contrast to this, a single-input, single-output control system has only one open-loop control signal, and consequently all the system state variable time responses depend on one desired system output response.

Also, notice that the maximum net reactivity insertion into the system is approximately 6¢ at time t equal to 20 seconds. At this same time, the corresponding external reactivity control signal is 40¢, and consequently the effect of the feedback loop is to subtract out 34¢ of reactivity worth. In other words, the prespecified steady-state value for the neutron power is too low for the magnitude of the reactivity insertion signal. Therefore, the control system must generate a strongly negative feedback signal to compensate for this imbalance and thereby reduce the electronic signal to the reactor control device. Without the feedback loop, the 2¢/second insertion signal for a period of 20 seconds would result in a power change much greater than the controlled response of Fig. 4-2.

Finally, linear values for the diode current I , diode voltage V , and diode electrical power IV at various times during the excursion are also compiled in Fig. 4-2. Note that the output voltage is within $\pm 5\%$ of the initial value, as hypothesized earlier.

Step change in Diode Load Resistance

The time responses given in Fig. 4-3 are the variations from the equilibrium point resulting from a step perturbation in external load resistance. The input perturbation $r_2(t)$ represents a $-0.007 \Omega \text{ cm}^2$ step change or a 20% drop in the external load resistance. The linear time responses for the controlled states, $x_1(t)$, $x_3(t)$, and $x_4(t)$, correspond to the transfer functions in the second column of the desired transfer matrix in Eq. (4-25). The fact that the linear response of the precursor concentration is approximately zero is verified by substituting the steady-state values for $\delta n(t)$ and $\delta p(t)$ into Eq. (4-6).

The effect of the added states of dynamic compensation is indicated by the plot of net diode load resistance $R_L(t)$ versus time. Although the $r_2(t)$ contribution to $u_2(t)$ is a negative step, $u_2(t)$ is a positive function of time. The reason is that although the fuel, emitter, and collector temperature states remain essentially at the equilibrium point during the first half second after the step change, the states of added compensation do not. In fact, positive step changes are induced in the dynamic compensator states by the negative change in $r_2(t)$. And since the states of added compensation are "fed back" to form the control signal $u_2(t)$, the net result is that $u_2(t)$ is positive

Fig. 4-3. Linear Responses for a $-0.007 \Omega\text{-cm}^2$ Step Change in Load Resistance

Fig. 4-3. Continued

t, sec	I, amps/cm ²	V, volts	IV, watts/cm ²
0.0	19.30	0.676	13.04
5.0	18.83	.690	12.99
10.0	18.86	.695	13.11
15.0	18.94	.697	13.20
20.0	19.00	.698	13.27
25.0	19.05	.699	13.31
30.0	19.08	.699	13.34
35.0	19.09	.699	13.35
40.0	19.10	.699	13.36
45.0	19.11	.699	13.37
50.0	19.11	.699	13.37
55.0	19.12	.699	13.37
60.0	19.12	.699	13.37

Fig. 4-3. Continued

and the net diode resistance increases to generate the desired output responses due to the control perturbation $r_2(t)$.

Fig. 4-3 also contains linear data for the diode current I , the diode voltage V , and the diode electrical power, IV , at various times during the system excursion. Again, the diode voltage variation is less than $\pm 5\%$ of the initial equilibrium value.

Ramp in Coolant Temperature

The time responses presented in Fig. 4-4 are the resultant system variations from equilibrium for a detected ramp rise in the bulk coolant temperature. The ramp input signal denoted $r_3(t)$ represents a $+1^\circ\text{C}/\text{second}$ increase in the coolant temperature for a time duration of 20 seconds. The linear time responses correspond to the transfer functions in the third column of the desired transfer matrix in Eq. (4-25).

The physical interpretation of the control signal $u_3(t)$ is that the coolant temperature is proportional to pumping power (assume the pumps are electromagnetic). Consequently, if there is a detected change in the electrical power delivered to the main pumps, a control pump must be activated such that the resultant coolant temperature varies as indicated by the plot of $u_3(t)$ versus time (Fig. 4-4). Without this compensating control pump, the resultant loss of power to the main pumps would induce a 20°C change (1°C per second for 20 seconds) in the primary coolant loop temperature. However, assuming the control pumps are present, the net change in T_{co} is limited to $+5^\circ\text{C}$ as indicated in Fig. 4-4 and the desired system output responses are achieved.

Fig. 4-4. Linear Responses for +1°C/sec Ramp in Coolant Temperature

Fig. 4-4. Continued

t, sec	I, amps/cm ²	V, Volts	IV, watts/cm ²
0.0	19.30	0.676	13.04
5.0	19.70	.671	13.23
10.0	20.41	.664	13.55
15.0	21.15	.656	13.87
20.0	21.90	.647	14.18
25.0	22.16	.645	14.29
30.0	22.05	.646	14.25
35.0	21.97	.647	14.22
40.0	21.92	.648	14.21
45.0	21.89	.648	14.20
50.0	21.88	.649	14.19
55.0	21.88	.649	14.19
60.0	21.88	.649	14.19

Fig. 4-4. Continued

Again, the linear values for diode current I , diode voltage V , and diode electrical power IV at various time intervals during the excursion are compiled in Fig. 4-4. The maximum deviation from the equilibrium diode voltage value is -4.6%.

Nonlinear Effects

The nonlinearities in the system come from the expressions for the neutron power density, the heat flux from the emitter $\delta q_E''$, and the diode electrical power $\delta(IV)$. The linearization of the neutron density, Eq. (4-9), involves neglecting the term $\delta\rho(t)\delta n(t)$ compared to $\delta n(t)$. Since the maximum value for $\delta\rho(t)$ from the time response curves is about 6ϕ , the error introduced by neglecting the $\delta\rho(t)\delta n(t)$ product is approximately a 6% underestimation of the peak neutron power density at time t equal to 20 seconds. At other times, the effect of the linearization is smaller, and consequently the linear results for the neutron power density should agree quite well with the nonlinear time responses.

The effect of the linearization for $\delta q_E''$ and $\delta(IV)$ given by Eqs. (4-18) and (4-19) should also be insignificant. This is due to the fact that the system variations for the emitter and collector temperatures and the diode load resistance are restricted to remain within the delta deviations used to compute the coefficients for the first-order Taylor expansions. To determine if the approximations for $\delta q_E''$ and $\delta(IV)$ are satisfactory, the integration routine was interrupted every 0.5 seconds, and the corresponding nonlinear values for $\delta q_E''$ and $\delta(IV)$ were computed with the SIMCON code for the emitter and collector temperatures and the diode load resistance at the time of interruption.

The results are plotted for each of the input perturbations in Figs. 4-5 and 4-6. The sporadic variations in the nonlinear values for $\delta q_E''$ and $\delta(IV)$ are due to the errors in the iteration process required to compute both diode current I and diode voltage V given a value of diode load resistance R_L . However, the curves indicate that the linear approximations for $\delta q_E''$ and $\delta(IV)$ are a satisfactory representation of the average nonlinear behavior for $\delta q_E''$ and $\delta(IV)$.

From the results presented above, it is expected that the linear time responses would represent within reasonable tolerances the corresponding nonlinear system time responses resulting from the set of input perturbations considered in this chapter.

Discussion of Results

The applicability of the design technique for an interacting multivariable control system is demonstrated on an in-core thermionic reactor system. The system representation is not a detailed description but is adequate for the purpose of an initial control-system study incorporating the interacting multivariable design technique. The concern was, therefore, not to develop a comprehensive mathematical model but to use a meaningful representation for the in-core thermionic reactor system and determine if an interacting control system could be designed which satisfied certain design constraints.

A set of design constraints were specified by hypothesizing that the thermionic reactor system was for a space power plant application. The limited set of constraints for the resultant control system consisted of 1) a constant diode voltage, 2) a thermal shock

Fig. 4-5. Nonlinear and Linear Responses of $\delta q_E''$

Fig. 4-6, Nonlinear and Linear Responses of $\delta(IV)$

limitation of less than $60^{\circ}\text{C}/\text{minute}$ in the collector and emitter temperatures, and 3) a separation of the thermal power and the diode electrical power dependency for sudden load changes. The three constraints are not hypothetical ones but may possibly be used in an actual control system design (Brehm, 1970).

The results of the time responses indicate that the linear responses do agree in all cases with the corresponding responses specified in the desired transfer matrix. Consequently, the F_T and G_T matrices generate the necessary compensation signals $u_1(t)$, $u_2(t)$ and $u_3(t)$ required to produce the interacting closed-loop responses specified by Eq. (4-25).

The design constraints were satisfied within certain tolerances. First, the nonlinear values for the diode voltage in Figs. 4-2, 4-3, and 4-4 indicate that the deviations from the equilibrium value for diode voltage are always less than $\pm 4.6\%$ during the system excursions. The steady-state values for diode voltage due to the input signals represent deviations of less than $\pm 4.0\%$ from V_0 . Also, the thermal shock constraint of $60^{\circ}\text{C}/\text{minute}$ on the emitter and collector temperatures was satisfied for all input perturbations, and the thermal power is decoupled from the diode electrical power fluctuations for sudden load changes as indicated in Fig. 4-3.

With regard to nonlinear closed-loop computations, initial investigations with the digital computer program employed by Guppy (1970) indicate that this program is not entirely satisfactory when the emitter and collector temperatures are restricted to small variations

about their equilibrium values (Note deviations in Fig. 4-6). However, when the variations are small, the linear representation must agree with the nonlinear and consequently the linear results are adequate.

CHAPTER 5

CONCLUSIONS

This chapter summarizes the contributions of this study and enumerates the results pertinent to the state variable feedback design of a multivariable control system possessing cross-coupling. Areas which require further research are also indicated.

Summary of Results

A feedback synthesis technique resulting in a closed-loop interacting transfer matrix containing desired transfer functions between all permutations of the system outputs and the external control inputs has been demonstrated. The feedback synthesis of a coupled multivariable system is based on previously established decoupling techniques for the state variable feedback design of noninteracting multivariable systems.

Specifically, a condensed review of the literature on decoupling techniques is presented which provides only the necessary information and formulation required to compute a pair of decoupling state-variable feedback and gain matrices for a multivariable system. A variational decoupling technique is developed which represents a systematic procedure for determining a set of state-variable feedback and gain matrices which will decouple the same multivariable system into any particular set of control input, system output relationships. The different input,

output relationships result by performing elementary row transformations on a reference vector-matrix representation.

By defining a particular elementary row transfer matrix, a restricted set of m decoupling matrices can be obtained which decouple each of the m control inputs with a different system output without repetition. The control-law matrices resulting in a closed-loop interacting system transfer matrix containing desired transfer functions for all possible combinations of system outputs and control inputs are formulated in terms of the set of m decoupling matrices. The formulation indicates that, in general, a closed form solution for the interacting control-law state feedback matrix does not exist because there are more unknown elements in the matrix than algebraic equations.

A method is demonstrated, however, whereby a satisfactory state feedback matrix can be computed by applying the Gaussian elimination process to the set of algebraic equations, reducing them to row echelon form. The coefficients of like powers of s are equated, and the elements in the feedback matrix are solved algebraically. It is shown that, in general, the resultant state feedback matrix for an interacting multi-variable control system will contain time dependent elements. This result leads to a new interpretation of the state variable control-law of Eq. (3-79). The simple product of the state feedback matrix and the state vector must now be expressed as a convolution integral over the product.

It is also shown, that if the desired transfer matrix is specified so that each row has transfer function elements containing the

same number of poles and zeros, identically located in the Laplace domain, the resultant state feedback matrix will have constant, time-independent elements. The gain constants of the transfer function elements in any particular row and the dynamics of any two rows do not have to be identical.

Finally, the interacting design technique is demonstrated on the in-core thermionic reactor model employed in the uncontrolled start-up analysis performed by Guppy (1970). Results indicate that it may be possible to use the technique to design an interacting control system for an in-core thermionic reactor which would satisfy a physically real set of control system constraints. However, the results obtained are applicable only to the single diode representation used to construct the mathematical model in this study. Further research is indicated in the next section which would represent a logical extension of the initial control study presented in Chapter 4.

Further Research

The development of the variational decoupling technique and the feedback synthesis technique for cross-coupled multivariable control systems presented in this study leave several unanswered questions and consequently establish areas for further investigation.

First, since the foundation for the design of an interacting closed-loop control system depends on state-variable decoupling techniques, what if the mathematical model for the system representation cannot be decoupled using state variable feedback (B^* singular)? Gilbert (1969) indicates that if the open-loop transfer matrix is nonsingular,

dynamic compensation can be added to the multivariable system to change B^* from a singular to a nonsingular matrix. However, the question of what type of compensation should be added is not answered. Consequently, one area of further research would be to formulate the required compensation in terms of specific system parameters so that B^{*-1} would exist for the augmented system.

The variational decoupling technique could possibly be applied to a multivariable system to determine the "best way" of decoupling the system when noninteraction is a design requirement. The criteria used to determine the "best way" might be the minimization of a set of sensitivity functions of the form, $s_{\lambda}^T = \frac{\lambda}{T} \frac{dT}{d\lambda}$ where T would equal the transfer function between the i^{th} system output and the j^{th} control input. Also, λ could be any variable parameter such as elements in the decoupling feedback matrix, open-loop system poles that may change with time, or possibly a particular class of control signals at the system inputs. The resultant vector-matrix representation resulting in the minimum set of sensitivity functions would then be the desired representation for the application of the decoupling techniques. The formulation of these ideas would constitute another area of further research.

The feedback synthesis technique also requires a prespecification of the desired transfer matrix. The method of choosing the desired transfer functions as illustrated in Chapter 4 is not the most desirable that can be envisioned. Instead, the possibility of applying optimization techniques such as the minimization of a particular performance index in specifying the desired system response might be

more feasible. This area should be investigated since the constraint on the type of transfer matrix that can be used for multivariable control systems is now removed.

Also, the feedback synthesis technique might prove extremely useful for the design of interacting control systems for open-loop plants that have inaccessible states. The reason is, that since the equation used to compute F_T [Eq. (3-90)] contains more unknowns than equations, the presence of inaccessible states would mean that those elements in the columns of F_T corresponding to the inaccessible states would be chosen a priori to be zero. For example, in the F_T matrix computed for the thermionic reactor control system [Eq. (4-26)], there is no feedback signal from the state of average fuel temperature. Consequently, the fact that the fuel temperature may or may not be an inaccessible state does not affect the realization of the desired closed-loop transfer matrix of Eq. (4-25). The possibility of applying the interacting feedback synthesis technique to plants with inaccessible states should be investigated further since it may eliminate the need to construct state-estimators.

Finally, the thermionic control system study should be expanded to include a multiple-diode representation with a complete heat rejection system. Spatial effects and thermal time delays in the heat rejection system might be included. Further investigations should be directed to the nonlinear program used by Guppy (1970) to determine the applicability in predicting the controlled nonlinear responses of the reactor model. In particular, experimental data for the designed

control system in Chapter 4 applicable to the single diode model would be most helpful in correlating the linear results obtained in this study.

APPENDIX

D_k COMPUTER PROGRAM

The program evaluates the D_k matrices for $k = 1, 2, \dots, m$, where $D_k = T_k^{-1} [sI_n - \hat{A}_k - \hat{B}_k \hat{F}_k]^{-1} \hat{B}_k \hat{G}_k$. A subcalculation also determines the F_k matrices for $k = 1, 2, \dots, m$, where $F_k = F_1^* + B_1^{*-1} \hat{F}_k \hat{T}_k$. The program is written in Fortran IV language and requires approximately 110k of core storage to compile on a CDC-6400 computer. The large quantity of storage is required to store and hold the many matrices used during the calculation. The program can handle problems as large as 10 state variables ($n = 10$) with up to 10 system outputs or control inputs ($m = 10$). Computing times for a problem with $n = 10$ and $m = 4$ is approximately 50 seconds on the CDC-6400.

The program consists of the main program DSUBK plus 18 subroutines. Eleven of the 18 subroutines are standard IBM SSP library subroutines which are also written in Fortran IV. The eleven IBM SSP subroutines required are called; ARRAY, LOC, GMADD, PRQD, SMPY, XCPY, GMPRD, PADD, PSUB, PCLA, and PMPY. The first seven subroutines are used as supplied by IBM but the last four are modified to handle triply dimensioned matrix elements and consequently are listed in this appendix.

The input data which must be supplied by the user for each D_k calculation are the scalars, n , m , k , and p_i , d_i , and ℓ_i for $i = 1, 2, 3, \dots, m$, plus the system matrices $\hat{A}_k$, $\hat{B}_k$, $\hat{C}_k$, $\hat{F}_k$, $\hat{G}_k$, $\hat{T}_k$, $\hat{T}_k^{-1}$,

F_1^* , and B^{*-1} . The output data consists, in part, of D_k and F_k for each k , $k = 1, 2, \dots, m$.

The first READ statement inputs the values for the scalars, n , m , TOL , and k , respectively. TOL is an internal tolerance (usually 10^{-10}) used to ignore round-off errors in computed numbers. The second READ statement inputs the values for p_i , $i = 1, 2, \dots, m+1$ given by Eqs. (2-46) and (2-48). The scalar p_i is the number of poles in the i^{th} subsystem resulting from the canonical transformation of the ID system. The third READ statement inputs the d_i , $i = 1, 2, \dots, m$ values of (Eq. 2-14). The fourth READ statement inputs the ℓ_i , $i = 1, 2, \dots, m$ values of Eq. (2-49). The scalar ℓ_i represents the number of zeros in the i^{th} subsystem. The next nine READ statements input the data for the $\hat{A}_k$, $\hat{B}_k$, $\hat{C}_k$, $\hat{F}_k$, $\hat{G}_k$, T_k^{-1} , F_1^* , T_k , and B_1^{*-1} matrices successively. The matrices are read by rows, six elements per card, with each new row starting on a new data card. The complete set of D_k matrices are computed by successively stacking the data cards for each value of k .

A complete listing of the main program DSUBK, the non-IBM subroutines, CANRT, CMPRD, CPMPY, CMPRD, MCOF, PMPRD, PINV, and SHIFT, and the modified IBM SSP subroutines, PADD, PSUB, PCLA, and PMPY are provided on the following pages.

PROGRAM DSUBK (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

COMPUTES THE D-SUB K MATRICES USED TO COMPUTE THE INTERACTING A
FEEDBACK AND GAIN MATRICES, F-SUB T AND G-SUB T.

DIMENSION AHAT(13,13),BHAT(13,13),CHAT(13,13)
DIMENSION FHAT(13,13),GHAT(13,13),IP(13),ND(13),NNR(13)
DIMENSION THAT(13,13),PHAT(13,13),RHAT(13,13)
DIMENSION FC(13,13), SHAT(13,13), HINV(13,13)
DIMENSION SHAT(13,13), THAT(13,13)
DIMENSION D1(13,13,13), LD1(13,13), RN(13,13,13), LRN(13,13)
DIMENSION TI(13,13,13), LTN(13,13)
DIMENSION WHAT(13),RRO(13),CRO(13),POL(13)
DIMENSION RRD(13), CRD(13), RD(13), CD(13)
COMMON/LABEL/D3(13,13,13), LD3(13,13), D2(13,13,13), LD2(13,13),
LDN(13,13,13), LDN(13,13), SN(13,13,13), LSN(13,13)

READ INPUT DATA

1 READ(5,58) N,M,TOL,ICASE
IF(EOF,5) 57,2
2 CONTINUE
MM=M+1
NR=0
READ(5,59) (IP(II),II=1,MM)
DO 3 KK=1,M
3 NP=IP(KK)+NR
NT=IP(MM)
READ(5,59) (ND(II),II=1,M)
READ(5,59) (NNR(II),II=1,M)
DO 4 II=1,N
4 READ(5,60) (AHAT(II,JJ), JJ=1,N)
DO 5 II=1,N
5 READ(5,60) (BHAT(II,JJ), JJ=1,M)
DO 6 II=1,M

```

6 READ(5,60) (CHAT(II,JJ), JJ=1,N)
  DO 7 II=1,M
7 READ(5,60) (FHAT(II,JJ), JJ=1,M)
  DO 8 II=1,M
8 READ(5,60) (GHAT(II,JJ), JJ=1,M)
  DO 9 II=1,N
9 READ(5,60) (THAT(II,JJ), JJ=1,N)
  DO 10 II=1,M
10 READ(5,60) (FC(II,JJ), JJ=1,N)
  DO 11 II=1,N
11 READ(5,60) (SMAT(II,JJ), JJ=1,N)
  DO 12 II=1,M
12 READ(5,60) (BINV(II,JJ), JJ=1,M)

```

WRITE INPUT DATA

```

WRITE(6,61) N,M,TOL
DO 13 II=1,M
13 WRITE(6,62) II,IP(II),ND(II),NNR(II)
  WRITE(6,62) NM,NI
  WRITE(6,63)
  DO 14 II=1,N
14 WRITE(6,64) (AHAT(II,JJ), JJ=1,N)
  WRITE(6,65)
  DO 15 II=1,N
15 WRITE(6,64) (BHAT(II,JJ), JJ=1,M)
  WRITE(6,66)
  DO 16 II=1,M
16 WRITE(6,64) (CHAT(II,JJ), JJ=1,N)
  WRITE(6,67)
  DO 17 II=1,M
17 WRITE(6,64) (BINV(II,JJ), JJ=1,M)
  WRITE(6,68)
  DO 18 II=1,M
18 WRITE(6,64) (FHAT(II,JJ), JJ=1,N)

```

```

        WRITE(6,70)
        DO 19 II=1,M
19      WRITE(6,64) (GHAT(II,JJ), JJ=1,M)
        WRITE(6,70)
        DO 20 II=1,N
20      WRITE(6,64) (SHAT(II,JJ), JJ=1,N)
        WRITE(6,71)
        DO 21 II=1,N
21      WRITE(6,64) (THAT(II,JJ), JJ=1,N)
        WRITE(6,72)
        DO 22 II=1,M
22      WRITE(6,64) (FC(II,JJ), JJ=1,N)

```

C
C
C

FORM R-SUB 11 SUBMATRIX

```

        CALL ARRAY(2,N,N,13,13,AHAT,AHAT)
        CALL ARRAY(2,N,N,13,13,BHAT,BHAT)
        CALL ARRAY(2,N,N,13,13,FHAT,FHAT)
        CALL GHPRD(BHAT,FHAT,PHAT,N,N,N)
        CALL GHADD(AHAT,PHAT,RHAT,N,N)
        CALL SMPY(RHAT,-1.,RHAT,N,N,0)
        CALL ARRAY(1,N,N,13,13,RHAT,RHAT)
        WRITE(6,73)
        DO 23 II=1,N
23      WRITE(6,64) (RHAT(II,JJ), JJ=1,N)
        DO 25 II=1,NR
        DO 24 JJ=1,NR
        LD1(II,JJ)=1
24      D1(1,II,JJ)=RHAT(II,JJ)
        LD1(II,II)=2
25      D1(2,II,II)=1.

```

C
C
C

COMPUTE (R-SUB 11)**-1 SUBMATRIX

```

        CALL PINV(RN,LRN,D1,LD1,NR)

```



```

WRITE(6,74)
DO 26 II=1,NR
DO 26 JJ=1,NR
KT=LNR(II,JJ)
26 WRITE(6,64) (RN(KK,II,JJ), KK=1,KT)
WRITE(6,75)
NNR=NR+1
KT=LNR(NNR,NNR)
WRITE(6,64) (RN(KK,NNR,NNR), KK=1,KT)
IF(NT) 57,38,27

```

FORM R-SUB 22 SUBMATRIX

```

27 DO 29 II=1,NT
III=II+NR
DO 28 JJ=1,NT
JJJ=JJ+NR
LDI(II,JJ)=1
28 D1(1,II,JJ)=RHAT(III,JJJ)
LD1(II,II)=2
29 D1(2,II,II)=1.

```

COMPUTE (R-SUB 22)*#-1 SUBMATRIX

```

CALL PINV(TN,LTN,D1,LD1,NT)
WRITE(6,76)
DO 30 II=1,NT
DO 30 JJ=1,NT
KT=LTN(II,JJ)
30 WRITE(6,64) (TN(KK,II,JJ), KK=1,KT)
WRITE(6,77)
NNT=NT+1
KT=LTN(NNT,NNT)
WRITE(6,64) (TN(KK,NNT,NNT), KK=1,KT)

```

```

C
C      FOR R-SUB 21 SUBMATRIX
C
31 DO 32 II=1,NT
    DO 32 JJ=1,NR
        III=II+NR
32 PHAT(II,JJ)=RHAT(III,JJ)
C
C      FORM PRODUCTS -((R-SUB 22)**-1)*(R-SUB 21)*((R-SUB 11)**-1)
C
    CALL ARRAY(2,NT,NR,13,13,PHAT,PHAT)
    CALL SMPY(PHAT,-1.,PHAT,NT,NR,0)
    CALL ARRAY(1,NT,NR,13,13,PHAT,PHAT)
    CALL CMPRO(D1,LD1,NT,NR,RN,LRN,NR,NR,PHAT,NT,NR,1)
    CALL PMPRO(SN,LSN,NT,NR,TN,LTN,NT,NT,D1,LD1,NT,NR)
    WRITE(6,78)
    DO 33 II=1,NT
        DO 33 JJ=1,NR
            KT=LSN(II,JJ)
33 WRITE(6,64) (SH(KK,II,JJ), KK=1,KT)
C
C      COMPUTE BHAT*GHAT MATRIX
C
    CALL ARRAY(2,M,M,13,13,GHAT,GHAT)
    CALL GMPRO(BHAT,GHAT,PHAT,N,M,M)
    CALL ARRAY(1,N,M,13,13,PHAT,PHAT)
    WRITE(6,79)
    DO 34 II=1,N
34 WRITE(6,64) (PHAT(II,JJ), JJ=1,M)
C
C      COMPUTE PRODUCTS ((S*I-AHAT-BHAT*FHAT)**-1)*BHAT*GHAT
C
    CALL ARRAY(2,N,M,13,13,PHAT,PHAT)
    CALL XCPY(PHAT,SHAT,1,1,NR,M,N,M,0)
    CALL XCPY(PHAT,THAT,NR+1,1,NT,M,N,M,0)

```

```

CALL ARRAY(1,NR,M,13,13,SHAT,SHAT)
CALL ARRAY(1,NT,M,13,13,THAT,THAT)
CALL CMPRD(D1,LD1,NR,M,RN,LRN,NR,NR,SHAT,NR,M,0)
CALL CMPRD(D2,LD2,NT,M,SN,LSN,NT,NR,SHAT,NR,M,0)
CALL CMPRD(D3,LD3,NT,M,TN,LTN,NT,NT,THAT,NT,M,0)
DO 36 JJ=1,M
DO 35 II=1,NR
35 CALL PCLA(RN,LRN,II,JJ,D1,LD1,II,JJ)
DO 36 II=1,NT
CALL PMPY(TN,LTN,II,JJ,RN,LRN,NR+1,D3,LD3,II,JJ)
36 CALL PADD(SN,LSN,II,JJ,D2,LD2,II,JJ,TN,LTN,II,JJ)

      COMPUTE PRODUCTS (T**1)*((S*I-AHAT-BHAT*FHAT)**-1)*BHAT*GHAT
C
C
C
CALL ARRAY(2,N,M,13,13,THAT,TMAT)
CALL XCPY(TMAT,SHAT,1,1,M,NR,N,N,0)
CALL XCPY(TMAT,THAT,1,NR+1,N,NT,N,N,0)
CALL ARRAY(1,N,NR,13,13,SHAT,SHAT)
CALL ARRAY(1,M,NT,13,13,THAT,THAT)
CALL CMPRD(D1,LD1,M,M,RN,LRN,NR,M,SHAT,N,NR,1)
CALL CMPRD(D2,LD2,M,M,SN,LSN,NT,M,THAT,N,NT,1)
DO 37 II=1,M
DO 37 JJ=1,M
CALL PMPY(D3,LD3,II,JJ,D1,LD1,II,JJ,TN,LTN,NT+1,NT+1)
37 CALL PADD(DN,LDN,II,JJ,D3,LD3,II,JJ,D2,LD2,II,JJ)
CALL PHPY(DN,LDN,N+1,M+1,TN,LTN,NT+1,NT+1,RN,LRN,NR+1,NR+1)

      COMPUTE F-SUB K =(FSTAR + (HSTAR**1)*FHAT*(T**1))
C
C
C
CALL ARRAY(2,N,M,13,13,SMAT,SHAT)
CALL ARRAY(2,M,N,13,13,FC,FC)
CALL ARRAY(2,M,N,13,13,BINV,BINV)
CALL GMPRD(FHAT,SMAT,RHAT,M,N,N)
CALL GMPRD(BINV,RHAT,FHAT,M,M,N)
CALL GMADD(FC,FHAT,PHAT,M,N)

```

```

CALL ARRAY(1,M,N,13,13,PHAT,PHAT)
GO TO 39
38 CALL ARRAY(2,M,M,13,13,GHAT,GHAT)
CALL GMPRO(BHAT,GHAT,PHAT,PHAT,M,M,M)
CALL ARRAY(1,M,M,13,13,PHAT,PHAT)
CALL CMPRO(D1,LD1,N,M,RN,LRN,N,N,PHAT,N,M,0)
CALL CMPRO(DN,LDN,N,M,01,LD1,N,M,THAT,N,N,1)

```

FORM D-SUB K DETERMINANT

```

CALL PCLA(DN,LDN,N+1,M+1,RN,LRN,N+1,N+1)
CALL ARRAY(2,M,N,13,13,FC,FC)
CALL ARRAY(2,M,N,13,13,SMAT,SMAT)
CALL ARRAY(2,M,M,13,13,BINV,BINV)
CALL GMPRO(FHAT,SMAT,RHAT,M,M,N)
CALL GMPRO(BINV,RHAT,PHAT,M,M,N)
CALL GADD(FC,FHAT,PHAT,M,N)
CALL ARRAY(1,M,N,13,13,PHAT,PHAT)
39 WRITE(6,80) ICASE
DO 48 I=1,N
DO 40 JJ=1,M
KT=LDN(I,JJ)
40 WRITE(6,64) (D,(KK,II,JJ), KK=1,KT)
WRITE(6,81) ICASE
NI=N+1
KT=LDN(NN,MN)
WRITE(6,64) (DN(KK,NN,MN),KK=1,KT)
KT=LDN(MN,MN)

```

COMPUTE D-SUB K DETERMINANT ROOTS

```

DO 41 KK=1,KT
41 WHAT(KK)=DN(KK,NN,MN)
CALL PROD(WHAT,KT,RRD,CRD,POL,ID,TER)
DO 55 JJ=1,M

```

```

DO 55 II=1,N
  IDIX=-1
  KT=LDN(II,JJ)
42 DO 43 KK=1,KT
43 WHAT(KK)=DN(KK,II,JJ)
C
C      COMPUTE D-SUB K COFACTOR ELEMENT ROOTS AND CANCEL WITH ALL
C      EQUAL ROOTS IN D-SUB K DETERMINANT
C
  CALL PROD(WHAT,KT,RRO,CRO,POL,IR,IER)
  IF(IER-2) 44,54,49
44 DO 45 KK=1,ID
  RD(KK)=RRO(KK)
45 CD(KK)=CRO(KK)
  IRD=ID
  CALL CANRT(RRO,CRO,RD,CD,IR,IRD)
  WRITE(6,82) ICASE,II,JJ,IER
  WRITE(6,83) DN(KT,II,JJ)
  IF(IR) 55,46,47
46 WRITE(6,84)
  WRITE(6,85)
  GO TO 48
47 WRITE(6,86) (RRO(KK),KK=1,IR)
  WRITE(6,87) (CRO(KK),KK=1,IR)
48 WRITE(6,88)
  WRITE(6,89) (RD(KK),KK=1,IRD)
  WRITE(6,90) (CD(KK),KK=1,IRD)
  GO TO 55
49 DO 50 KK=1,KT
50 IF(ABS(DN(KK,II,JJ)).LT.(1.E-5)) DN(KK,II,JJ)=0.
51 IF(DN(KT,II,JJ)) 53,52,53
52 KT=KT-1
  LDN(II,JJ)=KT
  GO TO 51
53 IDIX=IDIX+1

```

```

      IF (INDEX) 42,42,54
54  WRITE(6,72) ICASE,II,JJ,IER
      WRITE(6,84)
      WRITE(6,85)
      WRITE(6,86)
      WRITE(6,91)
      WRITE(6,92)
55  CONTINUE
      WRITE(6,93) ICASE
      DO 56 II=1,M
56  WRITE(6,64) (PHAT(II,JJ),JJ=1,N)
      GO TO 1
57  CONTINUE
58  FORMAT(2I3,E8.3,I2)
59  FORMAT(10(2X,I3))
60  FORMAT(6(E11.4,2X))
61  FORMAT(5X,3HN =,I2,5X,3HM =,I2,5X,16HZERO TOLERANCE =,E8.1,/)
62  FORMAT(5X,3HI =,I2,5X,9HP-SUB-I =,I2,5X,9HD-SUB-I =,I2,5X,9HR-SUB-
      II =,I2,/)
63  FORMAT(///,25X,11HAHAT MATRIX,/)
64  FORMAT(6(5XE15.6),/)
65  FORMAT(///,25X,11HRHAT MATRIX,/)
66  FORMAT(///,25X,11HCHAT MATRIX,/)
67  FORMAT(///,25X,20HRSTAR-INVERSE MATRIX,/)
68  FORMAT(///,25X,11HFHAT MATRIX,/)
69  FORMAT(///,25X,11HGHAT MATRIX,/)
70  FORMAT(///,25X,8HT MATRIX,/)
71  FORMAT(///,25X,16HT-INVERSE MATRIX,/)
72  FORMAT(///,25X,24HFSTAR COMPENSATED MATRIX,/)
73  FORMAT(///,25X,11HRHAT MATRIX,/)
74  FORMAT(/,25X,9HRN MATRIX,/)
75  FORMAT(/,25X,24HDETERMINANT OF RN MATRIX,/)
76  FORMAT(/,25X,9HTN MATRIX,/)
77  FORMAT(/,25X,24HDETERMINANT OF TN MATRIX,/)

```

```

78 FORMAT(/,25X,9HSN MATRIX,/)
79 FORMAT(/,25X,16HHMAT*GHAT MATRIX,/)
80 FORMAT(/,25X,5HD-SUB,I2,7H MATRIX,/)
81 FORMAT(/,25X,20HDETERMINANT OF D-SUB,I2,7H MATRIX,/)
82 FORMAT(///,3X,6HD-SUB-,I2,2H (,I2,1H,,I2,51H)
1 IER=,I2,/)
83 FORMAT(3X,27HNUMERATOR GAIN CONSTANT =,2X,E15.6,/)
84 FORMAT(3X,32HNUMERATOR REAL ROOTS = NONE)
85 FORMAT(3X,32HNUMERATOR COMPLEX ROOTS = NONE)
86 FORMAT(3X,27HNUMERATOR REAL ROOTS =,6(2XE15.6))
87 FORMAT(3X,27HNUMERATOR COMPLEX ROOTS =,6(2XE15.6))
88 FORMAT(1H0)
89 FORMAT(3X,27HDENOMINATOR REAL ROOTS =,6(2XE15.6))
90 FORMAT(3X,27HDENOMINATOR COMPLEX ROOTS =,6(2XE15.6))
91 FORMAT(3X,32HDENOMINATOR REAL ROOTS = NONE)
92 FORMAT(3X,32HDENOMINATOR COMPLEX ROOTS = NONE)
93 FORMAT(///,25X,5HF-SUB,I2,7H MATRIX,/)
STOP
END

```

```

SUBROUTINE PMPRD(Z, IDIMZ, IZ, JZ, X, IDIMX, IX, JX, Y, IDIMY, IY, JY)
DIMENSION Z(13,13,13), X(13,13,13), Y(13,13,13)
DIMENSION IDIMZ(13,13), IDIMX(13,13), IDIMY(13,13)

```

```

C
C      MULTIPLIES TWO MATRICES CONTAINING POLYNOMIAL ELEMENTS IN S
C

```

```

C      IF(JX-IY) 1,1,10

```

```

C
1 IF(IX-JX) 2,2,3
2 NNN=JX+1
  GO TO 4
3 NNN=IX+1
4 IF(JY-IY) 5,5,6
5 LLL=IY+1
  GO TO 7
6 LLL=JY+1
7 DO 10 LL=1,JY
  DO 10 NN=1,IX
    CALL PMPY(Z, IDIMZ, NNN, 1, X, IDIMX, NN, 1, Y, IDIMY, 1, LL)
    DO 9 KK=2,IY
      CALL PMPY(Z, IDIMZ, NNN, KK, X, IDIMX, NN, KK, Y, IDIMY, KK, LL)
      KKK=KK-1
    CALL PADD(Z, IDIMZ, NNN, LLL, Z, IDIMZ, NNN, KK, Z, IDIMZ, NNN, KKK)
    IF(KK-IY) 8,9,9
8 CALL PCLA(Z, IDIMZ, NNN, KK, Z, IDIMZ, NNN, LLL)
9 CONTINUE
  CALL PCLA(Z, IDIMZ, NN, LL, Z, IDIMZ, NNN, LLL)
10 CONTINUE
  RETURN
  END

```



```

SUBROUTINE CMPRD(Z, IDIMZ, IZ, JZ, X, IDIMX, IX, JX, Y, IY, JY, MS)
  DIMENSION Z(13,13,13), X(13,13,13), Y(13,13)
  DIMENSION IDIMZ(13,13), IDIMX(13,13)

```

```

      MULTIPLIES TWO MATRICES ONE CONTAINING POLYNOMIAL ELEMENTS
      IN S AND THE OTHER CONTAINING CONSTANT SCALAR ELEMENTS

```

```

      IF (MS) 1,1,7

```

```

      MS=0 MEANS POSTMULTIPLY X WITH Y
      MS=1 MEANS PREMULTIPLY X WITH Y

```

```

1  IF (IX-JX) 3,3,2

```

```

2  NNN=IX+1
   GO TO 4

```

```

3  NNN=JX+1

```

```

4  IF (JY-IY) 5,5,6

```

```

5  LLL=IY+1
   GO TO 13

```

```

6  LLL=JY+1
   GO TO 13

```

```

7  IF (IY-JY) 9,9,8

```

```

8  NNN=IY+1
   GO TO 10

```

```

9  NNN=JY+1

```

```

10 IF (JX-IX) 11,11,12

```

```

11 LLL=IX+1
   GO TO 18

```

```

12 LLL=JX+1
   GO TO 18

```

```

13 IF (JX-IY) 17,14,17

```

```

14 DO 17 LL=1,JY

```

```

DO 17 NN=1,IX
CALL CPMPY(Z,IDIMZ,NNN,1,X,IDIMX,NN,1,Y,1,LL)
DO 16 KK=2,IY
CALL CPMPY(Z,IDIMZ,NNN,KK,X,IDIMX,NN,KK,Y,KK,LL)
KKK=KK-1
CALL PADD(Z,IDIMZ,NNN,LLL,Z,IDIMZ,NNN,KK,Z,IDIMZ,NNN,KKK)
IF(KK-IY) 15,16,16
15 CALL PCLA(Z,IDIMZ,NNN,KK,Z,IDIMZ,NNN,LLL)
16 CONTINUE
CALL PCLA(Z,IDIMZ,NN,LL,Z,IDIMZ,NNN,LLL)
17 CONTINUE
GO TO 23
18 IF(JY-IX) 22,19,22
19 DO 22 LL=1,IY
DO 22 NN=1,JX
CALL CPMPY(Z,IDIMZ,NNN,1,X,IDIMX,1,NN,Y,LL,1)
DO 21 KK=2,IX
CALL CPMPY(Z,IDIMZ,NNN,KK,X,IDIMX,KK,NN,Y,LL,KK)
KKK=KK-1
CALL PADD(Z,IDIMZ,NNN,LLL,Z,IDIMZ,NNN,KK,Z,IDIMZ,NNN,KKK)
IF(KK-IX) 20,21,21
20 CALL PCLA(Z,IDIMZ,NNN,KK,Z,IDIMZ,NNN,LLL)
21 CONTINUE
CALL PCLA(Z,IDIMZ,LL,NN,Z,IDIMZ,NNN,LLL)
22 CONTINUE
23 RETURN
END

```

```
SUBROUTINE CPMPY(Z,IDIMZ,IZ,JZ,X,IDIMX,IX,JX,Y,IY,JY)
DIMENSION Z(13,13,13), X(13,13,13), Y(13,13)
DIMENSION IDIMZ(13,13), IDIMX(13,13)
```

C
C
C

MULTIPLIES A POLYNOMIAL BY A SCALAR

```
ID=0
IDX=IDIMX(IX,JX)
DO 1 IT=1,IDX
1 Z(IT,IZ,JZ)=X(IT,IX,JX)*Y(IY,JY)
2 IDT=IDX-ID
IF(Z(IDT,IZ,JZ)) 4,3,4
3 ID=ID+1
IF(IDT-1) 4,4,2
4 IDIMZ(IZ,JZ)=IDT
RETURN
END
```

```

SUBROUTINE PINV(COF,LCOF,D,LD,NDIM)
  DIMENSION DUM2(13,4,4), LDUM2(4,4), DUM3(13,5,5), LDUM3(5,5)
  DIMENSION DUM4(13,6,6), LDUM4(6,6), DUM5(13,7,7), LDUM5(7,7)
  DIMENSION DUM6(13,8,8), LDUM6(8,8), DUM7(13,9,9), LDUM7(9,9)
  DIMENSION DUM8(13,10,10), LDUM8(10,10)
  COMMON/LABEL/ D(13,13,13), LD(13,13), COF(13,13,13), LCOF(13,13),
  10DET(13,13,13), LDET(13,13), DUM1(13,13,13), LDUM1(13,13)

```

```

      PROGRAM COMPUTES THE INVERSE OF A MATRIX CONTAINING POLYNOMIAL
      ELEMENTS.  INVERSE COMPUTED USING THE ADJOINT MATRIX AND
      THE DETERMINANT

```

```

      TEST DIMENSION OF INPUT MATRIX

```

```

      GO TO (90,80,3,4,5,6,7,8,9,120)  NDIM

```

```

3  NR3=3
   GO TO 10
4  NR4=4
   NR3=1
   GO TO 20
5  NR5=5
   NR4=1$NR3=1
   GO TO 30
6  NR6=6
   NR5=1$NR4=1$NR3=1
   GO TO 40
7  NR7=7
   NR6=1$NR5=1$NR4=1$NR3=1
   GO TO 50
8  NR8=8
   NR7=1$NR6=1$NR5=1$NR4=1$NR3=1
   GO TO 60
9  NR9=9
   NR8=1$NR7=1$NR6=1$NR5=1$NR4=1$NR3=1

```

```

        GO TO 70
120 NR10=10
    NR3=1$NR4=1$NR5=1$NR6=1$NR7=1$NR8=1$NR9=1
    GO TO 170
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      10X10 MATRIX
C
170 DO 177 J10=1,NR10
    DO 177 I10=1,10
    CALL MCOF(DET,LODET,10,9,D,LD,I10,J10,0,13,13)
    CALL SHIFT(D,DUM8,LD,LDUM8,I10,J10,10,13,10)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      9X9 MATRIX
C
70 DO 77 J9=1,NR9
    DO 77 I9=1,9
    IF(NR9-1) 72,72,71
71 CALL MCOF(DET,LODET,9,8,D,LD,I9,J9,0,13,13)
    CALL SHIFT(D,DUM7,LD,LDUM7,I9,J9,9,13,9)
    GO TO 60
72 IF(LDUM8(I9,J9)-1) 100,972,973
972 IF(DUM8(1,I9,J9)) 973,974,973
974 LOET(I9,9)=1
    DET(1,I9,9)=0.
    GO TO 975
973 CALL MCOF(DET,LODET,9,8,DUM8,LDUM8,I9,J9,0,13,10)
    CALL SHIFT(DUM8,DUM7,LDUM8,LDUM7,I9,J9,9,10,9)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      8X8 MATRIX
C
60 DO 67 J8=1,NR8
    DO 67 I8=1,8
    IF(NR8-1) 62,62,61
61 CALL MCOF(DET,LODET,8,7,D,LD,I8,J8,0,13,13)
    CALL SHIFT(D,DUM6,LD,LDUM6,I8,J8,8,13,8)

```

```

        GO TO 50
    62 IF (LDUM7(I8,J8)-1) 100,962,963
    962 IF (DUM7(1,I8,J8)) 963,964,963
    964 LDET(I8,8)=1
        DET(1,I8,8)=0.
        GO TO 965
    963 CALL MCOF(DET,LDET,8,7,DUM7,LDUM7,I8,J8,0,13,9)
        CALL SHIFT(DUM7,DUM6,LDUM7,LDUM6,I8,J8,8,9,8)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      7X7 MATRIX
C
    50 DO 57 J7=1,NR7
        DO 57 I7=1,7
            IF (NR7-1) 52,52,51
    51 CALL MCOF(DET,LDET,7,6,D,LD,I7,J7,0,13,13)
        CALL SHIFT(D,DUM5,LD,LDUM5,I7,J7,7,13,7)
        GO TO 40
    52 IF (LDUM6(I7,J7)-1) 100,952,953
    952 IF (DUM6(1,I7,J7)) 953,954,953
    954 LDET(I7,7)=1
        DET(1,I7,7)=0.
        GO TO 955
    953 CALL MCOF(DET,LDET,7,6,DUM6,LDUM6,I7,J7,0,13,8)
        CALL SHIFT(DUM6,DUM5,LDUM6,LDUM5,I7,J7,7,8,7)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      6X6 MATRIX
C
    40 DO 47 J6=1,NR6
        DO 47 I6=1,6
            IF (NR6-1) 42,42,41
    41 CALL MCOF(DET,LDET,6,5,D,LD,I6,J6,0,13,13)
        CALL SHIFT(D,DUM4,LD,LDUM4,I6,J6,6,13,6)
        GO TO 30
    42 IF (LDUM5(I6,J6)-1) 100,942,943
    942 IF (DUM5(1,I6,J6)) 943,944,943

```

```

944 LDET(I4,6)=1
    DET(1,16)=0.
    GO TO 945
943 CALL MCOF(DET,LDET,6,5,DUM5,LDUM5,I6,J6,0,13,7)
    CALL SHIFT(DUM5,DUM4,LDUM5,LDUM4,I6,J6,6,7,6)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      5X5 MATRIX
C
30 DO 37 J5=1,NR5
    DO 37 I5=1,5
    IF(NR5-1) 32,32,31
31 CALL MCOF(DET,LDET,5,4,D,D,LD,I5,J5,0,13,13)
    CALL SHIFT(D,DUM3,LD,LDUM3,I5,J5,5,13,5)
    GO TO 20
32 IF(LDUM4(I5,J5)-1) 100,932,933
932 IF(DUM4(1,I5,J5)) 933,934,933
934 LDET(I5,5)=1
    DET(1,I5,5)=0.
    GO TO 935
933 CALL MCOF(DET,LDET,5,4,DUM4,LDUM4,I5,J5,0,13,6)
    CALL SHIFT(DUM4,DUM3,LDUM4,LDUM3,I5,J5,5,6,5)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      4X4 MATRIX
C
20 DO 27 J4=1,NR4
    DO 27 I4=1,4
    IF(NR4-1) 22,22,21
21 CALL MCOF(DET,LDET,4,3,D,D,LD,I4,J4,0,13,13)
    CALL SHIFT(D,DUM2,LD,LDUM2,I4,J4,4,13,4)
    GO TO 10
22 IF(LDUM3(I4,J4)-1) 100,922,923
922 IF(DUM3(1,I4,J4)) 923,924,923
924 LDET(I4,4)=1
    DET(1,I4,4)=0.
    GO TO 925

```

```

923 CALL MCOF (DET,LDET,4,3,DUM3,LDUM3,I4,J4,0,13,5)
    CALL SHIFT (DUM3,DUM2,LDUM3,LDUM2,I4,J4,4,5,4)
C      ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
C      3X3 MATRIX
C

```

```

    10 DO 17 J3=1,NR3
        DO 17 I3=1,3
            IF (NR3-1) 12,12,11
    11 CALL MCOF (DET,LDET,3,1,D,LD,I3,J3,0,13,13)
        CALL SHIFT (D,DUM1,LD,LDUM1,I3,J3,3,13,13)
        GO TO 13
    12 IF (LDUM2(I3,J3)-1) 100,912,913
912 IF (DUM2(1,I3,J3)) 913,914,913
914 LDET(I3,3)=1
    DET(1,I3,3)=0.
    GO TO 915
913 CALL MCOF (DET,LDET,3,1,DUM2,LDUM2,I3,J3,0,13,4)
    CALL SHIFT (DUM2,DUM1,LDUM2,LDUM1,I3,J3,3,4,13)
    13 CALL PMPY (DET,LDET,1,1,DUM1,LDUM1,1,1,DUM1,LDUM1,2,2)
    CALL PMPY (DET,LDET,2,2,DUM1,LDUM1,2,1,DUM1,LDUM1,1,2)
    CALL PSUB (DUM1,LDUM1,I3,J3,DET,LDET,1,1,DET,LDET,2,2)
    IF (NR3-1) 15,15,14
    14 CALL MCOF (COF,LCOF,J3,I3,DUM1,LDUM1,I3,J3,1,13,13)
    IF (J3-1) 15,15,17
    15 CALL PMPY (DET,LDET,I3,3,DUM1,LDUM1,I3,1,DET,LDET,3,1)
915 IF (I3-1) 17,17,16
    16 I13=I3-1
    CALL PADD (DET,LDET,2,1,DET,LDET,I3,3,DET,LDET,I13,3)
    CALL PCLA (DET,LDET,I3,3,DET,LDET,2,1)
    17 CONTINUE

```

```

C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C      3X3 MATRIX
C

```

```

    IF (NR3-1) 23,23,100
23 IF (NR4-3) 25,25,24

```



```

24 CALL MCOF(COF,LCOF,J4,I4,DET,LDET,3,3,1,13,13)
   IF(J4-1) 25,25,27
25 CALL PMPY(DET,LDET,I4,4,DET,LDET,3,3,DET,LDET,4,3)
925 IF(I4-1) 27,27,26
26 II4=I4-1
   CALL PADD(DET,LDET,4,1,DET,LDET,I4,4,DET,LDET,II4,4)
   CALL PCLA(DET,LDET,I4,4,DET,LDET,4,1)
27 CONTINUE
C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C      4X4 MATRIX
C
   IF(NR4-1) 33,33,100
33 IF(NR5-4) 35,35,34
34 CALL MCOF(COF,LCOF,J5,I5,DET,LDET,4,4,1,13,13)
   IF(J5-1) 35,35,37
35 CALL PMPY(DET,LDET,I5,5,DET,LDET,4,4,DET,LDET,5,4)
935 IF(I5-1) 37,37,36
36 II5=I5-1
   CALL PADD(DET,LDET,5,1,DET,LDET,I5,5,DET,LDET,II5,5)
   CALL PCLA(DET,LDET,I5,5,DET,LDET,5,1)
37 CONTINUE
C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C      5X5 MATRIX
C
   IF(NR5-1) 43,43,100
43 IF(NR6-5) 45,45,44
44 CALL MCOF(COF,LCOF,J6,I6,DET,LDET,5,5,1,13,13)
   IF(J6-1) 45,45,47
45 CALL PMPY(DET,LDET,I6,6,DET,LDET,5,5,DET,LDET,6,5)
945 IF(I6-1) 47,47,46
46 II6=I6-1
   CALL PADD(DET,LDET,6,1,DET,LDET,I6,6,DET,LDET,II6,6)
   CALL PCLA(DET,LDET,I6,6,DET,LDET,6,1)
47 CONTINUE
C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A

```

```

C      7X7 MATRIX
C
      IF (NR6-1) 53,53,100
53  IF (NR7-6) 55,55,54
54  CALL MCOF(COF,LCOF,J7,I7,DET,LDET,6,6,1,13,13)
      IF (J7-1) 55,55,57
55  CALL PMPY(DET,LDET,I7,7,DET,LDET,6,6,DET,LDET,7,6)
955 IF (I7-1) 57,57,56
56  II7=I7-1
      CALL PADD(DET,LDET,7,1,DET,LDET,I7,7,DET,LDET,II7,7)
      CALL PCLA(DET,LDET,I7,7,DET,LDET,7,1)
57  CONTINUE
C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C      7X7 MATRIX
C
      IF (NR7-1) 63,63,100
63  IF (NR8-7) 65,65,64
64  CALL MCOF(COF,LCOF,J8,I8,DET,LDET,7,7,1,13,13)
      IF (J8-1) 65,65,67
65  CALL PMPY(DET,LDET,I8,8,DET,LDET,7,7,DET,LDET,8,7)
965 IF (I8-1) 67,67,66
66  II8=I8-1
      CALL PADD(DET,LDET,8,1,DET,LDET,I8,8,DET,LDET,II8,8)
      CALL PCLA(DET,LDET,I8,8,DET,LDET,8,1)
67  CONTINUE
C      COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C      8X8 MATRIX
C
      IF (NR8-1) 73,73,100
73  IF (NR9-8) 75,75,74
74  CALL MCOF(COF,LCOF,J9,I9,DET,LDET,8,8,1,13,13)
      IF (J9-1) 75,75,77
75  CALL PMPY(DET,LDET,I9,9,DET,LDET,8,8,DET,LDET,9,8)
975 IF (I9-1) 77,77,76
76  II9=I9-1

```

CALL PADD(DET,LDET,9,1,DET,LDET,I9,9,DET,LDET,II9,9)

CALL PCLA(DET,LDET,I9,9,DET,LDET,9,1)

77 CONTINUE

COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
9X9 MATRIX

IF(NR9-1) 173,173,100

173 IF(NR10-9) 175,175,174

174 CALL MCOF(COF,LCOF,J10,I10,DET,LDET,9,9,1,13,13)

IF(J10-1) 175,175,177

175 CALL PMPY(DET,LDET,I10,10,DET,LDET,9,9,DET,LDET,10,9)

1975 IF(I10-1) 177,177,176

176 I110=I10-1

CALL PADD(DET,LDET,10,1,DET,LDET,I10,10,DET,LDET,I110,10)

CALL PCLA(DET,LDET,I10,10,DET,LDET,10,1)

177 CONTINUE

COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
10X10 MATRIX

GO TO 100

ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A
2X2 MATRIX

80 CALL PCLA(COF,LCOF,1,1,D,LD,2,2)

CALL PCLA(COF,LCOF,2,2,D,LD,1,1)

CALL MCOF(COF,LCOF,2,1,D,LD,2,1,13,13)

CALL MCOF(COF,LCOF,1,2,D,LD,1,2,13,13)

CALL PMPY(DET,LDET,1,1,D,LD,1,1,D,LD,2,2)

CALL PMPY(DET,LDET,2,2,D,LD,1,2,D,LD,2,1)

CALL PSUB(COF,LCOF,3,3,DET,LDET,1,1,DET,LDET,2,2)

COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
2X2 MATRIX

GO TO 110

ENTRY HERE TO COMPUTE ADJOINT MATRIX AND DETERMINANT OF A

```

C          1X1 MATRIX
C
90 CALL PCLA(COF,LCOF,2,2,D,LD,1,1)
   LCOF(1,1)=1
   COF(1,1,1)=1.
C          COMPLETE COMPUTATION OF ADJOINT MATRIX AND DETERMINANT FOR A
C          1X1 MATRIX
C
   GO TO 110
100 CALL PCLA(COF,LCOF,NDIM+1,NDIM+1,DET,LDET,NDIM,NDIM)
110 CONTINUE
   RETURN
   END

```

```

SUBROUTINE SHIFT(R,S,LR,LS,I,J,NRR,IM,IN)
DIMENSION R(13,IM,IM), S(13,IN,IN), LR(IM,IM), LS(IN,IN)

```

```

      FORMS A TIGHT (N-1) SUBMATRIX FROM THE RESULTANT MATRIX
      REMAINING FROM THE ROW AND COLUMN DELETION IN COMPUTING THE
      ADJOINT MATRIX

```

```

      NN=I+J
      NNN=2*NRR
      IF(NNN-NN) 17,17,1
1  IF(NRR-I) 13,13,2
2  IF(NRR-J) 9,9,3
3  JC=J
      MM=NRR-JC
      JJ=NRR-I
      DO 6 LL=1,MM
      JCC=JC+1
      IR=I
      DO 5 KK=1,JJ
      IRR=IR+1
      LS(IR,JC)=LR(IRR,JCC)
      LDIMS=LS(IR,JC)
      DO 4 K=1,LDIMS
      S(K,IR,JC)=R(K,IRR,JCC)
4  CONTINUE
5  IR=IR+1
6  JC=JC+1
7  IF(NN-2) 19,19,8
8  IF(J-1) 13,13,9
9  IR=I
      MM=NRR-IR
      JJ=J-1
      DO 11 LL=1,MM
      IRR=IR+1

```

```

      DO 10 KK=1,JJ
      LS(IR,KK)=LR(IRR,KK)
      LDIMS=LS(IR,KK)
      DO 10 K=1,LDIMS
      S(K,IR,KK)=R(K,IRR,KK)
10  CONTINUE
11  IR=IR+1
      IF(NRR-J) 16,16,12
12  IF(I-1) 19,19,13
13  JC=J
      JJ=NRR-JC
      MM=I-1
      DO 15 LL=1,JJ
      JCC=JC+1
      DO 14 KK=1,MM
      LS(KK,JC)=LR(KK,JCC)
      LDIMS=LS(KK,JC)
      DO 14 K=1,LDIMS
      S(K,KK,JC)=R(K,KK,JCC)
14  CONTINUE
15  JC=JC+1
      IF(J-1) 19,19,17
16  IF(I-1) 19,19,17
17  MM=I-1
      JJ=J-1
      DO 18 KK=1,MM
      DO 18 LL=1,JJ
      LS(KK,LL)=LR(KK,LL)
      LDIMS=LS(KK,LL)
      DO 18 K=1,LDIMS
      S(K,KK,LL)=R(K,KK,LL)
18  CONTINUE
19  RETURN
      END

```

```
SUBROUTINE MCOF(S,LS,IS,JS,R,LR,IR,JR,MS,IM,IN)  
DIMENSION R(13,IN,IN), S(13,IM,IM), LR(IN,IN), LS(IM,IM)
```

```
      C  
      C      DETERMINES THE SIGN OF THE COFACTOR AND PERFORMS THE  
      C      TRANSPOSE OPERATION  
      C
```

```
      IF(MS) 1,1,2
```

```
      C  
      1 LEX=IR+JR  
      GO TO 3  
      2 LEX=IS+JS  
      3 LX=LR(IR,JR)  
      LS(IS,JS)=LX  
      IF (LX) 5,5,4  
      4 DO 5 KT=1,LX  
      S(KT,IS,JS)=((-1.0)**LEX)*R(KT,IR,JR)  
      5 CONTINUE  
      RETURN  
      END
```

```

SUBROUTINE CANRT(ARN,ACN,ARD,ACD,IA,IIA)
DIMENSION ARN(1),ACN(1),ARD(1),ACD(1)

```

```

PROGRAM CANCELS ROOTS COMMON TO BOTH THE DETERMINANT AND EACH
COFACTOR ELEMENT IN THE ADJOINT MATRIX TRANSPOSED

```

```

TEST TO DETERMINE IF ANY CANCELLATION EXISTS DUE TO
PRESENCE OF NO ROOTS IN INPUT MATRIX

```

```

IF(IA) 14,14,1

```

```

1 KT=0
  KK=1
  KKD=1
2 IF(DABS(ARN(KK)-ARD(KKD)).LT.(1.D-03)) 3,12
3 IF(DABS(ACN(KK)-ACD(KKD)).LT.(1.D-03)) 4,12
4 IND=IIA-KKD
  IF(IND) 14,7,5
5 DO 6 II=1,IND
  II=II+KKD-1
  IIT=IT+1
  ARD(IIT)=ARD(IIT)
6 ACD(IIT)=ACD(IIT)
7 IIA=IIA-1
  IDD=IA-KK
  IF(IDD) 14,10,8
8 DO 9 II=1,IDD
  IS=II+KK-1
  IIS=IS+1
  ARN(IS)=ARN(IIS)
9 ACN(IS)=ACN(IIS)
10 IA=IA-1
  IF(KK-IA) 11,11,14
11 KKD=1

```



```
      KI=KT+1  
      GO TO 2  
12    KKD=KKD+1  
      IF (KKD-11A) 2,2,13  
13    KK=KK+1  
      KKD=1  
      KI=KT+1  
      IF (KK-1A) 2,2,14  
14    RETURN  
      END
```

```

SUBROUTINE PADD(Z, IDIMZ, IZ, JZ, X, IDIMX, IX, JX, Y, IDIMY, IY, JY)
  DIMENSION Z(50,6,6), X(50,6,6), Y(50,6,6)
  DIMENSION IDIMZ(6,6), IDIMX(6,6), IDIMY(6,6)
  DOUBLE PRECISION Z,X,Y

```

C
C
C

TEST DIMENSIONS OF SUMMANDS

```

  ID=0
  IOX=IDIMX(IX,JX)
  IOY=IDIMY(IY,JY)
  NDIM=IOX
  IF(IOX-IDY) 10,20,20
10 NDIM=IDY
20 IF(NDIM) 90,90,30
30 DO 80 IT=1,NDIM
  IF(IT-IDX) 40,40,60
40 IF(IT-IDY) 50,50,70
50 Z(IT,IZ,JZ)=X(IT,IX,JX)+Y(IT,IY,JY)
  GO TO 80
60 Z(IT,IZ,JZ)=Y(IT,IY,JY)
  GO TO 80
70 Z(IT,IZ,JZ)=X(IT,IX,JX)
80 CONTINUE
90 IDT=NDIM-ID
  IF(Z(IDT,IZ,JZ)) 110,100,110
100 ID=ID+1
  IF(IDT-1) 110,110,90
110 IDIMZ(IZ,JZ)=IDT
  RETURN
  END

```

```
SUBROUTINE PCLA(Y,IDIMY,IY,JY,X,IDIMX,IX,JX)  
  DIMENSION X(50,6,6), X(50,6,6), IDIMX(6,6), IDIMY(6,6)  
  DOUBLE PRECISION X,Y
```

C

```
  IDX=IDIMX(IX,JX)  
  IDIMY(IY,JY)=IDX  
  IF (IDX) 30,30,10  
10 DO 20 IT=1,IDX  
20 Y(IT,IY,JY)=X(IT,IX,JX)  
30 RETURN  
  END
```

```

SUBROUTINE PSUB(Z, IDIMZ, IZ, JZ, X, IDIMX, IX, JX, Y, IDIMY, IY, JY)
  DIMENSION Z(50,6,6), X(50,6,6), Y(50,6,6)
  DIMENSION IDIMZ(6,6), IDIMX(6,6), IDIMY(6,6)
  DOUBLE PRECISION Z,X,Y

```

C
C
C

TEST DIMENSIONS OF SUMMANDS

```

  ID=0
  IDX=IDIMX(IX,JX)
  IDY=IDIMY(IY,JY)
  NDIM=IDX
  IF (IDX-IDY) 10,20,20
10 NDIM=IDY
20 IF (NDIM) 90,90,30
30 DO 80 IT=1,NDIM
  IF (IT-IDX) 40,40,60
40 IF (IT-IDY) 50,50,70
50 Z(IT,IZ,JZ)=X(IT,IX,JX)-Y(IT,IY,JY)
  GO TO 80
60 Z(IT,IZ,JZ)=-Y(IT,IY,JY)
  GO TO 80
70 Z(IT,IZ,JZ)=X(IT,IX,JX)
80 CONTINUE
90 IDT=NDIM-ID
  IF (Z(IDT,IZ,JZ)) 110,100,110
100 ID=ID+1
  IF (IDT-1) 110,110,90
110 IDIMZ(IZ,JZ)=IDT
  RETURN
  END

```

```

SUBROUTINE PMPY(Z, IDIMZ, IZ, JZ, X, IDIMX, IX, JX, Y, IDIMY, IY, JY)
  DIMENSION Z(50,6,6), X(50,6,6), Y(50,6,6)
  DIMENSION IDIMZ(6,6), IDIMX(6,6), IDIMY(6,6)
  DOUBLE PRECISION Z,X,Y

```

C

```

  ID=0
  IDX=IDIMX(IX,JX)
  IDY=IDIMY(IY,JY)
  IF(IDX*IDY) 10,10,20
10 IDZ=0
  GO TO 50
20 IDZ=IDX+IDY-1
  DO 30 IT=1,IDZ
30 Z(IT,IZ,JZ)=0.
  DO 40 IT=1,IDX
  DO 40 JT=1,IDY
  KT=IT+JT-1
40 Z(KT,IZ,JZ)=X(IT,IX,JX)*Y(JT,IY,JY)+Z(KT,IZ,JZ)
50 IDT=IDZ-ID
  IF(Z(IDT,IZ,JZ)) 70,60,70
60 ID=ID+1
  IF(IDT-1) 70,70,50
70 IDIMZ(IZ,JZ)=IDT
  RETURN
  END

```

REFERENCES

- Anderson, B. D. O. and D. G. Luenberger, "Design of Multivariable Feedback Systems," Proc. of IEEE, Vol. 114, March, 1967.
- Asseo, A. J., "Phase-Variable Canonical Transformation of Multicontroller Systems," IEEE Trans. on Automatic Control, Vol. AC-13, February, 1968.
- Birta, L. G. and I. H. Mufti, "Some Results on an Inverse Problem in Multivariable Systems," IEEE Trans. on Automatic Control, Vol. AC-12, February, 1967.
- Bongiorno, J. J., Jr., "Minimum Sensitivity Design of Linear Multivariable Feedback Control Systems by Matrix Spectral Factorization," IEEE Trans. on Automatic Control, Vol. AC-14, December, 1969.
- Brasch, F. M., Jr. and J. B. Pearson, "Pole Placement Using Dynamic Compensators," IEEE Trans. on Automatic Control, Vol. AC-15, February, 1970.
- Brehm, R. L., Private Communication, November, 1970.
- Brehm, R. L. and D. L. Hetrick, J. G. Guppy, K. D. Kearns, T. R. Schmidt, "Dynamics of a Fast Reactor with In-Core Thermionic Converters, Second Annual Report," Engineering Experiment Station, University of Arizona, Tucson, Arizona (1968).
- Brehm, R. L. and D. L. Hetrick, D. A. Ferg, J. G. Guppy, K. D. Kearns, T. R. Schmidt, S. Yee, "Dynamics of a Fast Reactor with In-Core Thermionic Converters, Third Annual Report," Engineering Experiment Station, University of Arizona, Tucson, Arizona (1970).
- Brockett, R. W. and M. D. Mesarovic, "Synthesis of Linear Multivariable Systems," Trans. of AIEE (Applications in Industry), Vol. 81, September, 1962.
- Davison, E. J. and F. T. Man, "Interaction Index for Multivariable Control Systems," Proc. of IEE, Vol. 117, February, 1970.
- Falb, P. L. and W. A. Wolovich, "Decoupling in the Design and Synthesis of Multivariable Control Systems," IEEE Trans. on Automatic Control, Vol. AC-12, December, 1967.

- Frame, J. S., "Matrix Functions and Applications; Part I - Matrix Operations and Generalized Inverses," IEEE Spectrum, Vol. 1, March, 1964.
- Freeman, E. A. and K. M. Abbott, "Design of Optimal Multivariable Linear Control Systems," Proc. of IEE, Vol. 116, April, 1969.
- Gilbert, E. G., "Controllability and Observability in Multivariable Control Systems," SIAM Journal on Control, Vol. 1, No. 2, 1963.
- Gilbert, E. G., "The Decoupling of Multivariable Systems by State Feedback," SIAM Journal on Control, Vol. 7, No. 1, 1969.
- Gilbert, E. G. and J. R. Pivnichny, "A Computer Program for the Synthesis of Decoupled Multivariable Feedback Systems," IEEE Trans. on Automatic Control, Vol. AC-14, December, 1969.
- Guppy, J. G., "Analytical and Experimental Studies Relating to the Simulated Start-up of In-Core Thermionic Reactor Systems," PhD Dissertation, University of Arizona, Tucson, Arizona, June, 1970.
- Hetrick, D. L., Dynamics of Nuclear Reactors, University of Chicago Press, 1971.
- Hohn, F. E., Elementary Matrix Algebra, Macmillan Co., New York, Second Edition, 1966, pp. 140-141.
- Horowitz, I. M., "Synthesis of Linear Multivariable Feedback Control Systems," IRE Trans. on Automatic Control, Vol. AC-5, June, 1960.
- Kalman, R. E., "Mathematical Description of Linear Dynamical Systems," SIAM Journal on Control, Vol. 1, No. 2, 1963.
- Kavanagh, R. L., "Multivariable Control System Synthesis," Trans. of AIEE (Applications in Industry), Vol. 77, November, 1958.
- Krogh, F. T., "Variable Order Integrators for the Numerical Solution of Ordinary Differential Equations," JPL Sec. 314 (to be published), 1969.
- Kuo, B. C., Automatic Control Systems, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1962.
- Lindgren, A. G., "A Note on the Stability and Design of Interacting Multivariable Control Systems," IEEE Trans. on Automatic Control, Vol. AC-11, April, 1966.
- Luenberger, D. G., "Canonical Forms for Linear Multivariable Systems," IEEE Trans. on Automatic Control, Vol. AC-12, June, 1967.

- Morgan, B. S., "The Synthesis of Linear Multivariable Systems by State-Variable Feedback," IEEE Trans. on Automatic Control, Vol. AC-9, October, 1964.
- Mufti, I. H., "A Note on the Decoupling of Multivariable Systems," IEEE Trans. on Automatic Control, Vol. AC-14, August, 1969.
- Retallack, D. G., "Extended Root-Locus Technique for Design of Linear Multivariable Feedback Systems," Proc. of IEE, Vol. 117, March, 1970.
- Rosenbrock, H. H., "Design of Multivariable Control Systems Using the Inverse Nyquist Array," Proc. of IEE, Vol. 116, November, 1969.
- Schultz, D. G. and J. L. Melsa, State Functions and Linear Control Systems, McGraw-Hill Book Company, New York, 1967.
- Slivinsky, C. R., "The Design of Linear Multivariable Control Systems Using Modern Control Theory," PhD Dissertation, University of Arizona, Tucson, Arizona, 1969.
- Tyler, J. S. and F. B. Tuteur, "The Use of a Quadratic Performance Index to Design Multivariable Control Systems," IEEE Trans. on Automatic Control, Vol. AC-11, January, 1966.
- Weaver, L. E. and H. G. Gronroos, J. G. Guppy, J. P. Davis, "A Control Study for an In-Core Thermionic Reactor," JPL TR 32-1355, January, 1969.
- Wilkins, D. R., "SIMCON - A Digital Computer Program for Computing Thermionic Converter Performance Characteristics," GESR-2109, February, 1968.
- Wonham, W. M., "On Pole Assignment in Multi-Input Controllable Linear Systems," IEEE Trans. on Automatic Control, Vol. AC-12, December, 1967.