

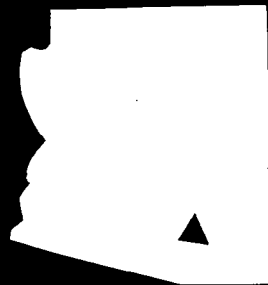
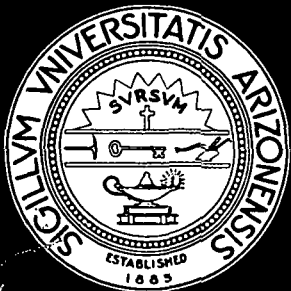
CASE FILE COPY

PRECOMPUTED STATE DEPENDENT DIGITAL CONTROL
OF A NUCLEAR ROCKET ENGINE

NASA (NGL 03-002-006)

SEMIANNUAL REPORT

JUNE 1972



*ENGINEERING EXPERIMENT STATION
COLLEGE OF ENGINEERING
THE UNIVERSITY OF ARIZONA
TUCSON, ARIZONA*

PRECOMPUTED STATE DEPENDENT DIGITAL CONTROL
OF A NUCLEAR ROCKET ENGINE

by
Morris Ray Johnson

NASA (NGL 03-002-006)
SEMIANNUAL REPORT
June 1972

Donald G. Schultz
Principal Investigator

TABLE OF CONTENTS

	Page
LIST OF ILLUSTRATIONS	v
LIST OF TABLES	viii
ABSTRACT	ix
1. INTRODUCTION	1
Description of NERVA	2
Review of NERVA Control Efforts	4
Organization	6
2. PRECOMPUTED STATE DEPENDENT DIGITAL CONTROL	9
Suboptimal Control Law Synthesis	
Background	10
Open and Closed-Loop Optimal Control	11
Control Law Synthesis Literature	
Review	12
Precomputed State Dependent Control	16
Mathematical Statement of PSDC	17
Off-Line Design Procedure	22
On-Line Operating Procedure	29
Examples of Precomputed State Dependent	
Digital Control	32
Linear System Example	32
Smith-Stenning Model Example	40
Summary	49
3. CONTROL MODEL DEVELOPMENT	53
The Common Analog Model	54
The Simplified Nonlinear Model	54
Control Model Derivation	57
Neutronics Approximation	58
Temperature Approximations	59
Transformation of Variables	61
Control Model Summary	61
Summary	63

TABLE OF CONTENTS--Continued

	Page
4. DESIGN OF THE CONTROL MODEL AND SIMPLIFIED NONLINEAR MODEL CONTROLLERS	65
Control Model Controller Off-Line Design	66
Nominal Trajectory Specification	66
Division of State Space	72
Block Geometry	75
Performance Index	78
CM Control Laws	79
Control Model Controller On-Line Operation	85
Simplified Nonlinear Model Controller Off-Line Design	87
Nominal Trajectory Specification	87
Performance Index	89
SNM Control Laws	90
Simplified Nonlinear Model Controller On-Line Operation	90
On-Line Estimation of B	90
Computation of DKD	94
DKD Magnitude and Rate Constraints	96
Summary	97
5. CONTROL MODEL AND SIMPLIFIED NONLINEAR MODEL CONTROL SYSTEM PERFORMANCE	99
Control Model Control System Performance	100
Nominal CM Control System Performance	100
CM Control System Recovery	102
CM Controller Sensitivity	104
Simplified Nonlinear Model Control System Performance	106
Nominal SNM Control System Performance	106
SNM Control System Recovery	109
SNM Controller Sensitivity	111
Summary	114
6. SUMMARY AND CONCLUSIONS	117
Summary	117
Conclusions	119
Suggestions for Further Study	121

TABLE OF CONTENTS--Continued

	Page
APPENDIX A: TEMPERATURE FUNCTION APPROXIMATIONS AND VARIABLE TRANSFORMATION IN THE CONTROL MODEL DEVELOPMENT	123
Temperature Function Approximations	123
Variable Transformation	127
APPENDIX B: COMPUTER PROGRAMS	135
Off-Line Optimization Program	136
Simplified Nonlinear Model Digital Simulation Program	144
REFERENCES	149

LIST OF ILLUSTRATIONS

Figure		Page
1-1	Pictorial Diagram of NERVA	3
2-1	Trajectory Specification Examples	20
2-2	Divergent Trajectory Specification for Linear System Example	34
2-3	Linear System Response with Divergent Trajectory Specification, $\underline{x}(0) = (0,1,5)$	36
2-4	Linear System Response with Divergent Trajectory Specification, $\underline{x}(0) = (3,1,5)$	37
2-5	Regulating Trajectory Specification for Linear System Example	38
2-6	Linear System Response with Regulating Trajectory Specification, $\underline{x}(0) = (-1.5,3,3)$	39
2-7	Smith-Stenning Model Trajectory Specification	43
2-8	Typical Block Geometry Used in Smith- Stenning Example Off-Line Optimizations . . .	44
2-9	Nominal Smith-Stenning Startup Time Response; $t_u = .08$	46
2-10	Nominal Smith-Stenning Startup Time Response; $t_u = .02$	47
2-11	Nominal Smith-Stenning Startup Time Response; $t_u = .32$	48
2-12	Smith-Stenning Startup Time Response with +150% Initial Error in P; $t_u = .08$	50
2-13	Smith-Stenning Startup Time Response with +50% Initial Error in C; $t_u = .08$	51

LIST OF ILLUSTRATIONS--Continued

Figure		Page
3-1	CM Controller State Vector Evolution	60
4-1	Normal NERVA Startup Temperature and Pressure Time Profiles	67
4-2	Partial State Space Representation of NERVA Engine Startup	68
4-3	Modified Trajectory Specification for CM Controller	70
4-4	Block Diagram of CM Control Update Sequence	86
4-5	SNM Control System Response to CM Controller, Block 4	88
4-6	SNM Control System Response to SNM Controller, Block 4	88
4-7	On-Line SNM Control Showing Conversion of Pseudo-Control S_d	95
5-1	CM Control System Startup/Shutdown Time Response	101
5-2	CM Control System Recovery Trajectories in the (T_{15}, B) Plane	103
5-3	CM Control System Recovery Trajectories in the (T_{15}, P_{15}) Plane	103
5-4	CM Control System Startup Trajectory Errors due to Changes in Turbopump Equation Coefficients	105
5-5	SNM Control System Startup/Shutdown Time Response	107
5-6	SNM Control System Recovery Trajectories in the (T_{15}, P_{15}) and (T_{15}, B) Planes	110
5-7	SNM Control System Startup B Trajectory and P_{15} Trajectory Error due to Changes in Turbopump Equation Coefficients	112

LIST OF ILLUSTRATIONS--Continued

Figure	Page
5-8 SNM Control System Startup B Trajectory and P_{15} Trajectory Error due to Changes in Temperature Equation Coefficients	113

LIST OF TABLES

Table		Page
4-1	CM and SNM State Space Block Boundaries	74
4-2	Optimization Point Coordinates for Block 4, CM and SNM Controllers	77
4-3	CM Control Law Tabulation	80
4-4	Contribution of CM Control Law Terms in Percent of Total; S Law Top, θ_{bcv} Law Bottom	84
4-5	SNM Control Law Tabulation	91

ABSTRACT

This study concerns a method of control which is applicable to multiple-input multiple-output nonlinear time-invariant systems in which desired behavior can be expressed explicitly as a trajectory in system state space. The method, called Precomputed State Dependent Control (PSDC), is proposed as a means of controlling a model of the NERVA nuclear rocket engine using an on-line digital computer.

The PSDC method is basically a synthesis technique in which a suboptimal control law is developed off-line, prior to system operation. This law is obtained by conducting searches at a finite number of points in state space, in the vicinity of some desired trajectory, to obtain a set of constant control vectors which tend to return the system to the desired trajectory. These vectors are used to evaluate the unknown coefficients in a control law having an assumed hyperellipsoidal form. The resulting coefficients constitute the heart of the controller and are used in the on-line computation of control vectors.

In the presentation, the PSDC method is stated in mathematical terms without reference to a particular application. Two examples of PSDC are given prior to the more detailed description of the NERVA control system development.

The most detailed model of NERVA is called the Common Analog Model (CAM). The Simplified Nonlinear Model (SNM), a lower order model developed from the CAM, provides the basis for this study. To facilitate application of the PSDC method to NERVA, a lower order Control Model is first derived from the SNM. Various aspects of off-line design and on-line operation of the SNM controller are examined, including computational requirements and control constraints.

Experiments are conducted to evaluate three aspects of the SNM control system performance: the ability of the system to follow the desired trajectory, its ability to return to this trajectory from distant initial conditions, and the sensitivity of the controller to variations in dominant plant time constants. These experiments indicate that PSDC is a practical and effective means of controlling the Simplified Nonlinear Model of NERVA. The SNM control system exhibits excellent trajectory following properties and is capable of returning to the desired trajectory from nearly any point in state space which is physically realizable. Also, the controller is reasonably insensitive to variations in plant time constants.

The main contributions of this study are the development of the Control Model, and the introduction of the PSDC technique. While the Control Model itself may find little application outside this or similar NERVA studies, the PSDC method is more generally applicable.

CHAPTER 1

INTRODUCTION

In this study a control method, termed Precomputed State Dependent Control, is developed in an effort to control a model of the NERVA nuclear rocket engine with an on-line digital computer. The method is more generally applicable to systems whose desired performance can best be described by some trajectory in its state space rather than by some other performance criterion such as time minimization. That is, the system is required to traverse some specified trajectory as closely as possible in spite of perturbations or changes in system characteristics.

Controller design is accomplished by synthesizing control laws off-line, prior to actual operation. Appropriate control vector values at a finite number of points in state space in the vicinity of the specified trajectory are computed, and then least squares fits are performed over this vector set. The resulting coefficients form the heart of the controller and are used in the on-line computation of control vectors.

The primary contribution of this work is the application of the Precomputed State Dependent Control (PSDC) method to a specific system. The system in question is the

NERVA nuclear rocket engine and is briefly described in the section which follows. This introductory chapter includes a review of various efforts which have been made toward controlling the NERVA system, and a complete organizational outline.

Description of NERVA

Historically the concept of nuclear propulsion dates back to 1959, with the KIWI proof-of-principle experiment (Crouch, 1965) marking the first serious attempt at building actual hardware. The most recent efforts are centered around NERVA (Nuclear Engine for Rocket Vehicle Application) with Aerojet Nuclear Systems Company as prime contractor and Westinghouse Corporation and Los Alamos playing important research roles.

The engine uses hydrogen as a coolant/propellant. Liquid H_2 is pumped from storage tanks through cooling jackets in the nozzle and support structure, into the reactor via a turbine, and finally expelled through the nozzle throat as a hot gas. The turbine is linked to a pump located at the H_2 storage tank exit with the turbopump powering the engine in a bootstrap arrangement. The system is thus characterized by the reactor neutronics, coolant flow dynamics, and various physical system constraints.

Fig. 1-1 is an engine diagram depicting its operation. Note that temperatures, hydrogen flow rates,

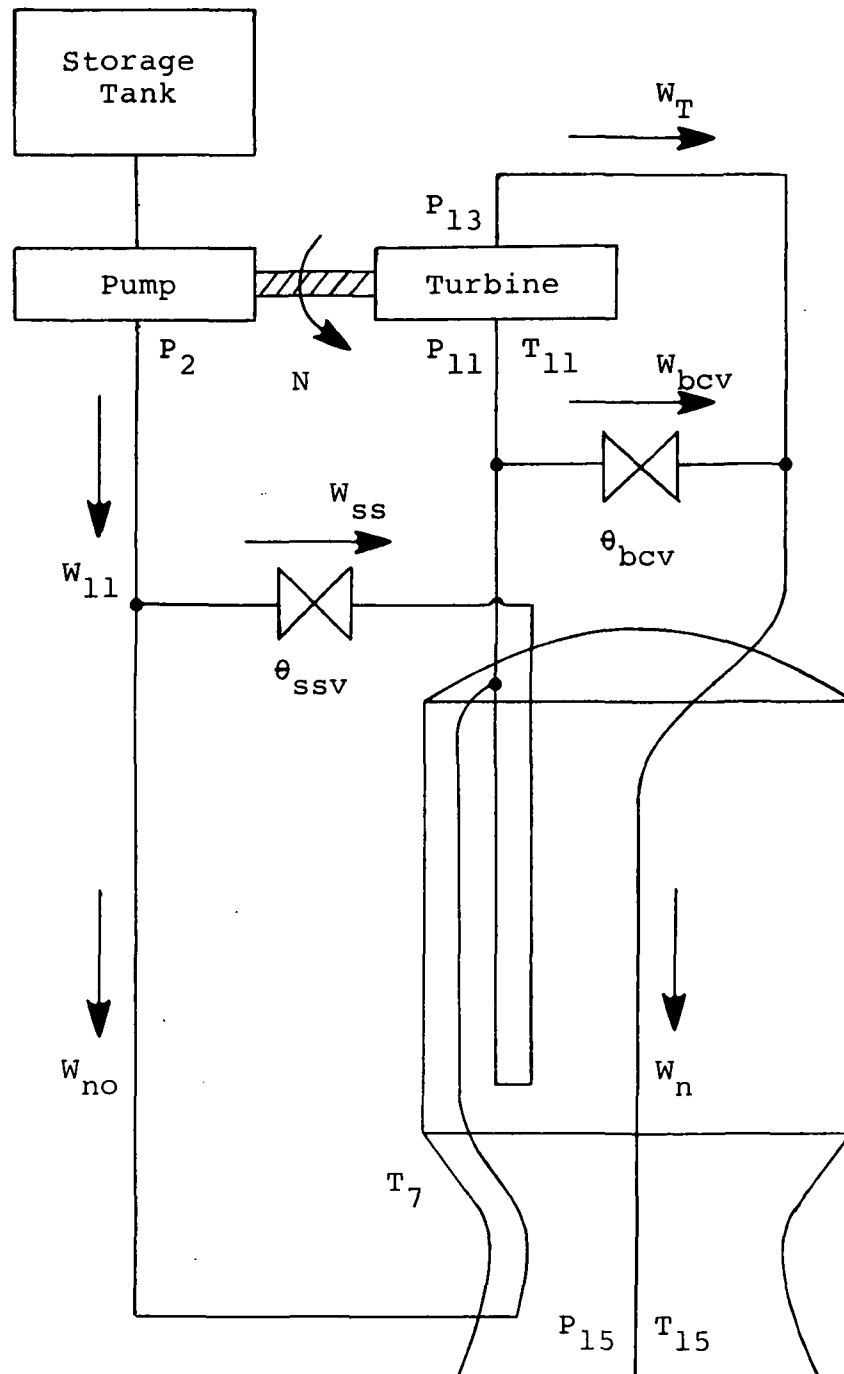


Fig. 1-1. Pictorial Diagram of NERVA

pressures, etc., are shown at various points. This nomenclature is used throughout the presentation so that the reader can refer to it for a better intuitive understanding of the system as the discussion develops.

Several mathematical models of the engine have been developed as interest in nuclear rocket propulsion has increased. Smith and Stenning (1961) have developed a simple model for conceptual studies, but this model is not sufficiently detailed to represent the present NERVA configuration. More recently, Aerojet and Westinghouse personnel have developed a very detailed model of NERVA called the Common Analog Model (CAM) which serves as a basis for system analyses. Kendrick (1972) has developed a simpler and lower order model, the Simplified Nonlinear Model (SNM), which retains the basic dynamic behavior of the CAM. The SNM is simple enough to provide at least a basis for controller design, and it is so used in this study.

Review of NERVA Control Efforts

Numerous articles appear in the literature concerning nuclear rocket engine control, stability, and optimization; e.g., Mohler (1966), Smith (1962), Smith and Stenning (1961). Those written on optimal control are highly idealized and are invariably limited to open-loop control. While these studies are important from a conceptual standpoint, they are not an end in themselves; the goal is ultimately

closed-loop control. As regards the more practical control and stability studies, while some do consider the closed-loop case, these studies usually rely on linear control theory. This approach is valid for systems containing only mild nonlinearities and for which desired system performance can be specified in terms of such classical quantities as frequency response, phase and gain margins, and so forth. The NERVA engine does not fall into this category. Here, nonlinearities could never be described as mild and, due to some rather stringent physical constraints, ideal performance cannot be described by classical s-domain quantities alone.

More recent NERVA control efforts emphasizing digital controllers fall into two categories. The first, the approach being considered by Aerojet Nuclear Systems, can be termed Precomputed Time Dependent Control. It is based on a detailed off-line examination of the control vector time sequence required to drive the system along a certain desired pressure-temperature profile, and the storing of this information for use on-line. This method essentially constitutes open-loop control with a limited closed-loop correction capability.

The second approach is that taken by Kendrick (1972) called State Dependent State Variable Feedback. Here the digital controller samples the system state, linearizes an assumed model about that state, computes the linear feedback

coefficients necessary to reduce this model to one of low order having desirable pole-zero locations, and finally computes the resulting control vector. This procedure is repeated at a fixed rate. No off-line computation is necessary and the on-line requirements are proportional to the complexity of the linearized state dependent model used to generate control vectors. This approach cleverly provides "digital control" while still retaining an s-domain design philosophy.

Although these two control methods have been used with some success, several aspects of control performance are lacking. The NERVA system is such that ideal behavior is best described by a state space trajectory in which engine efficiency (the total impulse realized from a given weight of expended propellant) is balanced against some critical physical constraints. As previously stated, it is this type of performance specification for which the PSDC method is suited. The purpose of this study, then, is to design a digital controller using the PSDC method which forces the Simplified Nonlinear Model of NERVA along a near-ideal state space trajectory.

Organization

This section contains a brief organizational outline of the entire presentation. Chapter 2 is basically a mathematical statement of the Precomputed State Dependent

Control method. Since this method is a type of control law synthesis, it is of interest to review the literature for similar methods and applications. Chapter 2 includes such a review. Many aspects of PSDC controller design are discussed, including trajectory specification, control constraints, and various on and off-line computational requirements. Also, two examples of this control technique are presented.

Chapter 3 covers the development of the Control Model, derived from the SNM, which is subsequently used in the SNM controller design. Chapter 4 describes the actual controller design using the theory presented in Chapter 2 and the Control Model developed in Chapter 3. The SNM state space trajectory specification is discussed in detail along with many computational aspects of the problem.

All numerical results describing the SNM control system performance are contained in Chapter 5. The system is examined to determine its ability to follow the trajectory specification, its ability to return to this trajectory when perturbed, and its sensitivity to changes in plant parameters. Experiments show good results in all three control characteristics. As a matter of interest and as a source of comparison, parallel experiments are conducted on the control system composed of the Control Model itself and its corresponding PSDC controller. The results of these experiments indicate that the Control Model is a valid

representation of the Simplified Nonlinear Model, and verify that Precomputed State Dependent Control is a practical and effective means of controlling the SNM.

Chapter 6 concludes the study and summarizes the important contributions therein. Suggestions for further study in this area are also included.

CHAPTER 2

PRECOMPUTED STATE DEPENDENT DIGITAL CONTROL

Several schemes for controlling the NERVA nuclear rocket engine system have been proposed. The two most notable and recent proposals are reviewed in Chapter 1 and are not discussed further here except to point out the main limitations of each. The precomputed time dependent control method proposed by Aerojet Nuclear Systems suffers from the necessity of storing time dependent control sequences and reference trajectories. This requires considerable on-line storage. Also, the nature of this time dependent control is such that changes in plant parameters may invalidate the control sequence. The on-line control method proposed by Kendrick (1972) removes the large storage requirement at the expense of a more complex on-line computational requirement. He shows that the method is quite tolerant of variations in plant parameters; but the controller, even without parameter variations, allows the plant to deviate considerably from the reference trajectory.

The Precomputed State Dependent Control method (PSDC) is proposed as an alternative to these two methods. The PSDC method has, potentially, a smaller on-line computation/storage requirement than either of the other two

methods, and exhibits excellent trajectory following properties. Also, since control is state (rather than time) dependent, system response is reasonably insensitive to plant parameter variations.

In this chapter the PSDC method is presented in mathematical terms together with a discussion of controller design and operation. Several examples of PSDC application are given. This presentation is preceded by a section giving the reader the necessary suboptimal control background. The PSDC method is basically a control law synthesis technique and this background includes a review of similar control methods appearing in the literature.

Suboptimal Control Law Synthesis Background

A large amount of effort is expended in the study of optimal control. However, most of these efforts are concerned with obtaining an open-loop control, $u^*(t)$, which extremizes some performance index. Theoretically, in fact, Pontryagin's Maximum Principle (Pontryagin et al., 1962) can be used to find $u^*(t)$ for a broad class of nonlinear systems and performance indices. In practice, solutions are difficult to obtain and are of questionable value. At present the only general method of obtaining an optimal closed-loop control law, $u^*(\underline{x})$, is through the use of Dynamic Programming. See Larson (1967) for a good survey of Dynamic

Programming techniques. The computational difficulties are still formidable, however, and increase exponentially with the order of the system in question.

This section introduces an alternative approach, that of generating or synthesizing a suboptimal control law. Absolute optimality is sacrificed in order to generate control laws more easily and in a form which can be practically implemented. First, open and closed-loop control is discussed in more detail, and then a number of synthesis techniques appearing in the literature are reviewed.

Open and Closed-Loop Optimal Control

Throughout the presentation, symbols which are not underlined are scalar quantities, underlined lower case symbols are vectors, and underlined upper case symbols are matrices. Consider a nonlinear system whose dynamics are described by the differential equation set,

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{u}, t) \quad (2-1)$$

where $\underline{x}$ is a state vector and $\underline{u}$ is a control vector. The optimal control problem is stated as follows. Find a control $\underline{u}^*(t)$ which minimizes a performance index,

$$J[\underline{u}(t)] = \int_0^T g[\underline{x}(t), \underline{u}(t), t] dt \quad (2-2)$$

subject to the differential constraint represented by (2-1).

By using variational calculus and solving the resulting two-point boundary value problem (TPBVP) it is possible, though difficult, to find $\underline{u}^*(t)$ in many cases. If $\underline{u}^*(t)$ is applied to (2-1), the system is driven along the optimal trajectory, $\underline{x}^*(t)$. Now $\underline{u}^*(t)$ constitutes open-loop control and is usually an inadequate means of control from a practical point of view. The optimality of $\underline{u}^*(t)$ is highly dependent upon changes in the initial value, $\underline{x}(0)$, the final time, T , and various system parameters.

The disadvantages of open-loop control suggest the development of a state dependent optimal control by eliminating t in the parametric representation, $\underline{u}^*(t)$, $\underline{x}^*(t)$, and giving $\underline{u}^*(\underline{x})$. Johnson and Gibson (1964) show that it is possible to generate this closed-loop optimal control exactly as a state function, only for a linear system and a quadratic performance index. For nonlinear systems and more general performance indices it is not possible to generate $\underline{u}^*(\underline{x})$ which exactly corresponds to $\underline{u}^*(t)$, $\underline{x}^*(t)$. It is possible, however, to generate a suboptimal closed-loop control $\underline{u}(\underline{x})$, by restricting its form to some easily implementable state function and employing one of several synthesis techniques recently developed.

Control Law Synthesis Literature Review

Efforts directed toward suboptimal control law synthesis currently appearing in the literature can be roughly

categorized into three areas. A few papers from these groups are referenced here to indicate the depth of research being conducted at this time.

The first area involves approximating the (previously found) optimal quantities $\underline{u}^*(t)$ and/or $\underline{x}^*(t)$ by some simple analytic time function. Typical of this approach is a method presented by Chang and DeRusso (1964) wherein $\underline{x}^*(t)$ is approximated by a truncated series of exponential functions. The resulting control is, by definition, open-loop and this area is not considered further.

The second area is typified by Kelley (1964) in which control is based on deviations of the system states from their optimal values. The control law is initially found by considering perturbations around an optimal trajectory which has previously been found. This technique and other similar ones e.g., Breakwell, Speyer, and Bryson (1963) and Levine (1966), all suffer the same disadvantage of having to store reference trajectories (functions of time) as a part of the controller. Note that the method of control being investigated by Aerojet Nuclear Systems for control of NERVA falls into this area.

The third area contains some methods which are more promising for practical control applications. An excellent paper by Longmuir and Bohn (1967) outlines a method for synthesizing suboptimal control laws. To begin, M optimal "development trajectories", $u_k^*(t)$, $k = 1, \dots, M$, are found

corresponding to M initial points within the region of interest. The form of the control law for each separate control is chosen to be

$$u(\underline{x}, \underline{c}) = \sum_{i=1}^N c_i z_i(\underline{x}) \quad (2-3)$$

where $\underline{z}(\underline{x})$ is an N -vector of "basis functions" and $\underline{c}$ is an N -vector of unknown coefficients. Now for the k th trajectory consider the quantity

$$J^k(\underline{c}) = \int_0^{T_k} \left(u[\underline{x}(t, \underline{x}_{ok}), \underline{c}] - u_k^*[t, \underline{x}_{ok}] \right)^2 dt \quad (2-4)$$

which is the mean square error between the exact and approximate controls along the k th trajectory. If the control law is to be based on only one development trajectory, then $\underline{c}$ is chosen so as to minimize (2-4). More likely the design is based upon a whole set of development trajectories, in which case $\underline{c}$ is chosen to minimize $J(\underline{c})$ below.

$$J(\underline{c}) = \sum_{k=1}^M J^k(\underline{c}) \quad (2-5)$$

The coefficient vector $\underline{c}$ is found algebraically because of the form of (2-5). This is essentially a trajectory matching process. Meyer (1971) uses a method very similar to

that of Longmuir and Bohn (1967), except that control laws are restricted to be piecewise-linear in $\underline{x}$.

Love and Lavi (1968) present a two-stage process for control law synthesis. The first stage enables the designer to choose the form of the control law, $\underline{u}(\underline{x})$, by considering a linear combination of all candidate basis functions and then systematically eliminating the unnecessary ones. Having arrived at the simplest form for $\underline{u}$,

$$\underline{u}(\underline{x}) = \underline{Bz}(\underline{x}) \quad (2-6)$$

the elements in the matrix $\underline{B}$ are chosen to minimize a very general performance index. That is, (2-6) is substituted into (2-1) and (2-2) giving

$$\begin{aligned} \dot{\underline{x}} &= \underline{f}'(\underline{x}, \underline{B}) \\ J(\underline{B}) &= \int_0^T g'[\underline{x}(t), \underline{B}, t] dt \end{aligned} \quad (2-7)$$

Since $\underline{B}$ is a constant matrix, the equation $\dot{\underline{B}} = 0$ is adjoined to (2-7) and the set becomes a TPBVP. Or, $\underline{B}$ may be solved by a direct minimization method such as one of the gradient search techniques. The methods described by Graham and Strauss (1970) and Eisenberg and Sage (1966) are similar, and contain additional means for selecting the form of $\underline{u}(\underline{x})$.

Finally Raju and Brueggeman (1970) present a method of synthesizing a suboptimal control law which is sensitive

to variations in plant parameters. The basic method is the same as that proposed by Longmuir and Bohn (1967), except that the coefficients associated with the various basis functions are not constant but are functions of the plant parameters themselves.

These methods are not intended to be used for problems in which the sign of a switching function determines the optimal control. Here it is better to approximate the switching surface itself as a state function. Papers involving such problems include Mendel and Zapalac (1967), Rooy (1970), and Luh and Shafran (1970).

Precomputed State Dependent Control

In the synthesis methods reviewed in the previous section, the desired (optimal) trajectory is expressed implicitly as a performance index. The Precomputed State Dependent Control method is proposed to obtain control laws for a slightly different class of problems in which the desired trajectory is available as an explicit function of the state vector, $\underline{x}$. It is shown in Chapter 4 that the nuclear rocket engine control problem falls into this class.

The general PSDC method is described in this section. The method is developed in mathematical terms without reference to any particular application. Implementation of this control scheme is divided into two phases. First, a controller is designed by performing certain numerical

computations. This is accomplished off-line digitally and is done only once. Second, the completed controller, an on-line digital computer programmed according to the off-line design results, actually controls the given plant. These two phases, off-line controller design and on-line operation, are discussed following the basic PSDC method description.

Mathematical Statement of PSDC

Consider a nonlinear time-invariant deterministic system whose dynamics are described by the differential equation set

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{u}) \quad (2-8)$$

where $\underline{x}$ is an n th order state vector and $\underline{u}$ is an m th order control vector with $m < n$. For a region r in state space let the desired trajectory (hereafter called the nominal trajectory) be defined by the p equations,

$$\begin{aligned} x_1 &= h_1(x_{p+1}) \\ &\vdots \\ x_i &= h_i(x_{p+1}) \\ &\vdots \\ x_p &= h_p(x_{p+1}) \end{aligned} \quad (2-9)$$

where h_i is a single-valued function for all i . Equations (2-9) merit some explanation. For an n th order system a

complete trajectory specification involving all n states is described by its projection into $n - 1$ orthogonal state space planes. That is, $p = n - 1$. Certain states may not be of direct interest and their trajectories need not be specified. In this case $p < n - 1$, and the trajectory is specified only in the p planes of interest corresponding to the p equations in (2-9). For convenience equations (2-9) can be expressed as $\underline{H}_r = 0$.

It should be noted that only p independent controls are required to drive a system along a trajectory which has been specified in p orthogonal planes. That is, $m = p$ even though p is less than $n - 1$. To show this, expand (2-8).

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_m) \end{aligned} \tag{2-10}$$

Differentiating (2-9),

$$\begin{aligned} \dot{x}_1 &= g_1(x_{p+1}, \dot{x}_{p+1}) \\ &\vdots \\ \dot{x}_p &= g_p(x_{p+1}, \dot{x}_{p+1}) \end{aligned} \tag{2-11}$$

and substituting (2-9) and (2-11) into (2-10),

$$\begin{aligned}
g_1 &= f_1(h_1, \dots, h_p, x_{p+1}, \dots, x_n, u_1, \dots, u_m) \\
&\vdots \\
g_p &= f_p(h_1, \dots, h_p, x_{p+1}, \dots, x_n, u_1, \dots, u_m) \\
\dot{x}_{p+1} &= f_{p+1}(h_1, \dots, h_p, x_{p+1}, \dots, x_n, u_1, \dots, u_m) \\
&\vdots \\
\dot{x}_n &= f_n(h_1, \dots, h_p, x_{p+1}, \dots, x_n, u_1, \dots, u_m)
\end{aligned} \tag{2-12}$$

Now (2-12) constitutes n equations with the number of unknowns equal to $n - p$ states plus m controls. Thus a unique solution for $\underline{u}$ theoretically exists only if $m = p$.

For notational convenience in this trajectory specification discussion, it is assumed that the first p states are expressed as functions of the one independent state, x_{p+1} . The basic PSDC method and the proof that $m = p$ are not limited to this case, however. For example the following specification involving two independent states is acceptable.

$$\begin{aligned}
x_1 &= h_1(x_2) \\
x_3 &= h_2(x_4)
\end{aligned} \tag{2-13}$$

The range of allowable trajectory specifications is quite broad but must, of course, be physically realizable. Also, in some cases one has no choice as to which direction the system will move along a given trajectory. Fig. 2-1 shows three example trajectories in the (x_1, x_2) plane with $x_2 = \dot{x}_1$. Trajectories A and B cannot be realized except at

the origin. Trajectory C is realizable, but for $x_1 > -1$ movement will be up and right, while for $x_1 < -1$ movement will be down and left.

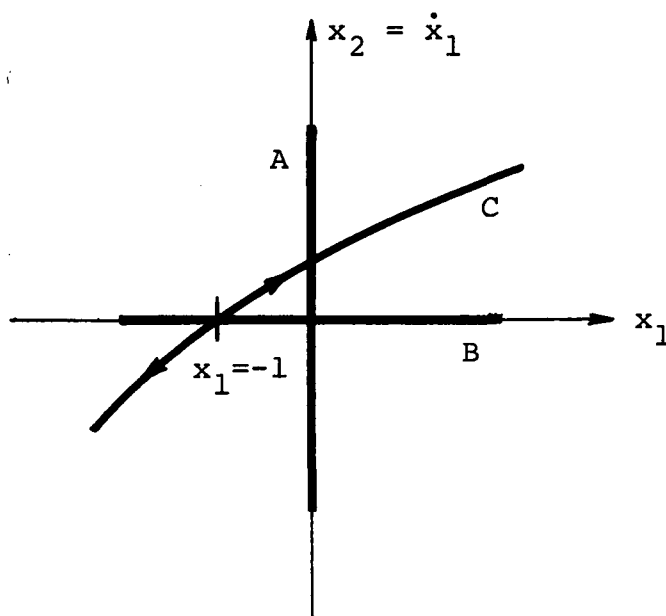


Fig. 2-1. Trajectory Specification Examples

With the system given by (2-8) and the trajectory specification (2-9) explained, the PSDC method is stated simply as follows. An approximate control law $\underline{u}(\underline{x})$ is obtained off-line by conducting searches at N_c points in region r to obtain a set of constant control vectors which tend to return the system from each of the N_c points to any other point on the nominal trajectory. These vectors are used to evaluate the unknown coefficients c_{ij} in an approximate control law of the form

$$u_j(\underline{x}) = \sum_{i=1}^N c_{ij} z_{ij} , \quad j = 1, \dots, m \quad (2-14)$$

These coefficients constitute the controller and are used in the on-line computation of control vectors. As the system moves through state space, the states are sampled and a new control vector is computed using equation (2-14). This is repeated at a rate fast enough to adequately approximate continuous control.

The PSDC method does not include any formal means for determining the "best" form for $\underline{u}(\underline{x})$. A few of the papers mentioned in the previous literature review contain suggestions for arriving at a suitable form and these are applicable here. For low-order systems it may be possible to find $\underline{u}^*(t)$ and $\underline{x}^*(t)$ corresponding to the nominal trajectory specification, and this may give some hint as to what form should be used for $\underline{u}(\underline{x})$. At this point in the discussion, however, it is assumed that a form is somehow chosen and can be expressed by equation (2-14).

The development of a controller for a very nonlinear system operating over wide ranges may require that the total applicable state space be divided into a number of regions or blocks, and a separate law found for each block. Such a controller is difficult to implement classically, but becomes an easy and straightforward matter if done with a digital computer.

Off-Line Design Procedure

The off-line or design phase of a PSDC application is discussed in this subsection. It is assumed that the system equations (2-8) and trajectory specification (2-9) are given. Basically the off-line procedure involves the synthesis of approximate control laws covering all applicable state space. In each block of state space, searches are conducted at N_c points to obtain a set of optimal constant control vectors which tend to force the system toward the nominal trajectory.

A numerical search or optimization implies the existence of a performance index which must be defined. Also, an analytical form for the approximate control law must be chosen. Once a form is chosen there remains the task of finding the control law coefficients which best correspond, in a least squares sense, to the set of N_c optimal controls. These aspects of PSDC controller design are considered here.

Performance Indices. Consider the problem of finding a constant control vector which drives the system from the point $\underline{x}$ to some point nearer the nominal trajectory. In terms of penalty or cost, this operation can be divided into two parts: the cost of moving from point $\underline{x}$ to some other point as the result of control $\underline{u}$ applied for dT seconds, and the cost of moving from there to some point on the nominal trajectory. The total cost is the sum of these two, and the optimal control $\underline{u}^*$ to be assigned to point $\underline{x}$ is that which

minimizes this sum. That is,

$$I_r(\underline{x}) = \min_{\underline{u}} \left\{ \int_0^{dT} L[\underline{H}_r, \underline{x}(t)] dt + I_r(\underline{x} + \underline{dx}) \right\} \quad (2-15)$$

where $\underline{x} + \underline{dx}$ is the intermediate state. This separation of cost is justified by the Principle of Optimality (Larson 1968, p. 16) which states, simply, that "Given an optimal trajectory from point A to point C, the portion of the trajectory from any intermediate point B to point C must be the optimal trajectory from point B to point C." Since $I_r(\underline{x} + \underline{dx})$ is not known it can be initially approximated and equation (2-15) solved iteratively for $\underline{u}^*$. This iterative method is called Bellman's Approximation in Function Space (Bellman and Dreyfus 1962).

Equation (2-15) contains a cost functional, L , which is simply a manifestation of penalty associated with points not on the nominal trajectory. Therefore,

$$L[\underline{H}_r, \underline{x}] \geq 0 \quad (2-16)$$

with equality whenever $\underline{x}$ lies exactly on the line $\underline{H}_r = 0$. The functional might be, for example, the orthogonal distance from the point $\underline{x}$ to the line $\underline{H}_r = 0$, or the euclidean norm of the pth order error vector,

$$E = \begin{bmatrix} e_1 \\ \vdots \\ e_p \end{bmatrix} = \begin{bmatrix} x_1 - h_1(x_{p+1}) \\ \vdots \\ x_p - h_p(x_{p+1}) \end{bmatrix} \quad (2-17)$$

An iterative solution to equation (2-15) can be avoided by choosing different forms for $I_r(\underline{x})$. Three are suggested here.

1. The second term in (2-15) is approximated by

$$I_r(\underline{x} + \underline{dx}) \approx L[\underline{H}_r, \underline{x}(dT)] \quad (2-18)$$

which gives

$$I_r(\underline{x}) = \min_{\underline{u}} \left\{ \int_0^{dT} L[\underline{H}_r, \underline{x}(t)] dt + L[\underline{H}_r, \underline{x}(dT)] \right\} \quad (2-19)$$

2. The second term in (2-15) is omitted, leaving

$$I_r(\underline{x}) = \min_{\underline{u}} \left\{ \int_0^{dT} L[\underline{H}_r, \underline{x}(t)] dt \right\} \quad (2-20)$$

The resulting $\underline{u}^*$ is one which minimizes deviations from the nominal trajectory during the period dT , but which offers no assurance of a small deviation at $t = dT$.

3. The first term in (2-15) is omitted, leaving

$$I_r(\underline{x}) = \min_{\underline{u}} \left\{ L[\underline{H}_r, \underline{x}(dT)] \right\} \quad (2-21)$$

The resulting $\underline{u}^*$ causes the trajectory to coincide with the nominal after dT seconds, but offers no assurance against large deviations in the open interval $0 < t < dT$.

In each case, $\underline{u}^*$ can be found in a single search. Each different performance index will naturally result in a different control law. All three indices mentioned above will, however, result in laws which have the same general tendency--to drive the system from point $\underline{x}$ toward the nominal trajectory.

Numerical Optimization Method. The off-line portion of a PSDC application centers around the procurement of N_c optimal control vectors associated with the N_c selected points in some region r . It is this vector set which ultimately defines the control law, $\underline{u}(\underline{x})$. Each m -dimensional search is made by attempting to minimize whatever performance index has been chosen.

Many different numerical schemes are available with which to conduct these searches. The so-called method of Parallel Tangents (Wilde and Beightler 1967, Harkins 1964) is used in the examples presented in this study. This method is characterized by fast convergence even when the function to be extremized contains "ridges". The method does not converge rapidly to the exact optimum as the optimum is approached closely, but an exact solution is not needed. It is hardly necessary to obtain an optimum correct to eight significant figures when the results are to be part of a

least squares fit which will be meaningful only to, say, three significant figures.

Control Law Form. As previously indicated, the PSDC method allows any $\underline{u}(\underline{x})$ which is of the form given in equation (2-14). To maintain some uniformity to the presentation a hyperellipsoidal form is used for $\underline{u}(\underline{x})$ in all the examples considered in this study. While other types of functions may have fewer terms and/or may describe the true optimal control function better for a particular system, all continuous control "surfaces" can be described by a hyperellipsoidal surface quite accurately if state space is subdivided finely enough. With this justification, all $\underline{u}(\underline{x})$ in this study are given the following form.

$$\begin{aligned} u_1(\underline{x}) &= b_1 + \underline{c}_1 \underline{x} + \underline{x}^T \underline{D}_1 \underline{x} \\ &\vdots \\ u_m(\underline{x}) &= b_m + \underline{c}_m \underline{x} + \underline{x}^T \underline{D}_m \underline{x} \end{aligned} \tag{2-22}$$

where b_i is a scalar, $\underline{c}_i$ is an n -vector, and $\underline{D}_i$ is an $n \times n$ symmetric matrix. There is some redundancy in $\underline{D}_i$ because of its symmetry. The number of coefficients required to describe each of m controls is

$$N = \frac{n^2 + 3n + 2}{2} \tag{2-23}$$

which is fewer than the $n^2 + n + 1$ indicated by equations

(2-22). As an example, for $n = 2$ and $m = 1$ the control law is assumed to fit the form,

$$u(\underline{x}) = a_1 + a_2 x_1 + a_3 x_2 + a_4 x_1^2 + a_5 x_1 x_2 + a_6 x_2^2 \quad (2-24)$$

Least Squares Fit Process. Once the functional form of $u(\underline{x})$ and a performance index are chosen, it is necessary to compute optimal control vectors at each of N_c points within region r . The coordinates of these points are chosen so as to obtain uniform coverage in the region and to obtain a pattern which does not contain an ill-conditioned coordinate matrix. For instance, if all points are nearly collinear, a nearly singular or ill-conditioned matrix will be encountered in the fit process.

To simplify the description of the fit process, consider the case where $m = 1$; that is, the single control is given by

$$u(\underline{x}) = \sum_{i=1}^N a_i z_i \quad (2-25)$$

Form the matrix equation,

$$\underline{F} \underline{a} = \underline{v} \quad (2-26)$$

where $\underline{F}$ is an $N_c \times N$ block coordinate matrix with f_{ij} the i th basis function, z_i , evaluated at the coordinates of the

jth optimization point, $\underline{a}$ is an Nth order coefficient vector and $\underline{v}$ is an N_c -vector containing the optimal values of control computed at each of the N_c points. If $N_c = N$, $\underline{F}$ is square and $\underline{a}$ is found by simply inverting $\underline{F}$.

$$\underline{a} = \underline{F}^{-1} \underline{v} \quad (2-27)$$

If $N_c > N$, premultiply (2-27) by $\underline{F}^T$.

$$\underline{F}^T \underline{F} \underline{a} = \underline{F}^T \underline{v} \quad (2-28)$$

Now $(\underline{F}^T \underline{F})$ is $N \times N$ and can be inverted, giving

$$\underline{a} = (\underline{F}^T \underline{F})^{-1} \underline{F}^T \underline{v} \quad (2-29)$$

which is equivalent to a least squares solution. Note that if $m > 1$, the quantity $(\underline{F}^T \underline{F})^{-1} \underline{F}^T$ is computed only once and used in all m control fits in block r.

$$\underline{a}^k = (\underline{F}^T \underline{F})^{-1} \underline{F}^T \underline{v}^k, \quad k = 1, \dots, m \quad (2-30)$$

Control Constraints. a control law which contains constraints may not be a smooth, well behaved function of state, but may have abrupt edges wherever the constraint is reached. A good hyperellipsoidal fit to this type of surface is not to be expected. One way to avoid this problem is by specifying a nominal trajectory which confines the system to regions of state space where the nominal control is unconstrained. In some cases, including the nuclear

rocket engine system emphasized in this study, this is done as a matter of course. If a controller is designed on this basis, there is some question as to how the system will respond from an initial point so far removed from the nominal trajectory that the controller prescribes a control which violates a constraint. With the control physically constrained, the system would not react in the normal manner. It would, however, almost surely tend to drive the system to a condition where the controller could return the system to the nominal trajectory with a normal, unconstrained, control. There seems to be no way of proving this type of recovery capability other than by experimentation.

Where desired trajectories are such that bang-bang control is called for, the problem should be reformulated so that the switching boundaries themselves are approximated as state functions.

On-Line Operating Procedure

This subsection deals with the on-line implementation of the suboptimal control law developed in the off-line design phase of a PSDC application. Operation of the digital controller is discussed as are means of approximating the necessary on-line storage and speed requirements.

Controller Operation. The nature of the PSDC method is such that a digital computer is best suited to implement the closed-loop control. A small special purpose computer

can be used, or an existing computer can be used on a time sharing basis. The actual controller consists of a memory containing the control law coefficients computed in the off-line procedure, and a program which executes the following steps sequentially and cyclically:

1. Sample the system states which are arguments of $\underline{u}(\underline{x})$.
2. Compute the corresponding block number.
3. Retrieve the applicable control coefficients if the block number has changed since the previous control update.
4. Compute the control vector from the appropriate expressions; i.e., equation (2-14).

These basic steps can be enlarged upon to fit the application. For example, step two can be expanded to provide a measure of system security by sensing large abnormal system perturbations. The above control vector update sequence is repeated at a rate whose minimum is related to the period dT used in the off-line computational performance index. This point is discussed in more detail in regard to the nuclear rocket engine application in Chapter 4. There is no upper limit on the control update rate other than that imposed by the on-line computer limitations.

Controller Requirements. It is difficult to make any quantitative statement concerning on-line computer requirements because they vary so widely from one application

to another. It is possible, however, to set certain guidelines as to the necessary storage and speed capabilities. Storage is required for the actual update program instructions and for the control law coefficients. Storage space required for instructions is difficult to estimate since it depends upon the complexity of the control law form and upon the particular computer to be used, but is quite minimal owing to the simplicity of the basic program. On the other hand, the necessary storage space for control law coefficients is readily found. If each of m control laws requires N coefficients and if applicable state space is divided into R blocks, then mNR words of storage are required.

Estimating the time required for each control update is not possible without referring to a particular digital computer. The update instructions are examined for the number of adds, multiplies, memory references, etc. Then, knowing the instructional performance of the particular machine, the total update time is found. For instance, the update instructions for the Smith-Stenning Model controller (example two in this chapter involving four states and two controls) requires 22 adds, 28 multiplies, and 44 memory references including I/O reference. Assuming 4 μ s per add, 20 μ s per multiply, and 2 μ s per memory reference (only modest digital capability), update time is 736 μ s. The update period in this example is 80 ms; thus the digital controller operates at less than one percent duty cycle. In any case,

the update time must obviously be less than the desired update period.

Examples of Precomputed State Dependent Digital Control

Two examples of Precomputed State Dependent Digital Control are presented in this section. Because the purpose of these examples is simply to illustrate the basic concept of PSDC, controller design and system performance are given only in brief. Chapters 4 and 5 cover the design and performance of two nuclear rocket engine controllers, and give sufficient detail to make the design process clear.

Linear System Example

The first example deals with a third-order linear system. By choosing a linear system and specifying linear nominal trajectories, an analytic solution for the optimal control $\underline{u}^*(t)$ is easily found and serves as a check on controller performance. The two-input system is described by the following state equations.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - x_2 - x_3 + u_1 \\ \dot{x}_3 &= x_2 - 2x_3 + u_2\end{aligned}\tag{2-31}$$

Since $m = 2$, the desired system trajectory may be specified in two planes; thus the trajectories of all three states are under complete control.

Two separate trajectory specifications are investigated. One forces the plant away from the origin and the other regulates about an equilibrium point. Fig. 2-2 depicts the "divergent" trajectory which is also stated in the following equations.

$$\begin{aligned}x_2 &= x_1 + 1 \\x_3 &= 5 - x_1\end{aligned}\tag{2-32}$$

The dotted lines in the figure show the boundaries of the single state space region to be processed, and the x 's indicate the points where off-line optimizations are performed. An optimal control vector is computed at each of the 64 points by minimizing the performance index,

$$J(\underline{u}) = L[\underline{H}, \underline{x}(0.1)]\tag{2-33}$$

where $L[\underline{H}, \underline{x}]$ is the orthogonal distance from point $\underline{x}$ to the nominal trajectory, $\underline{H} = 0$. The two control laws are obtained by performing least squares quadratic fits over the 64 values, assuming the standard hyperellipsoidal form for both laws. The results are shown below.

$$\begin{aligned}u_1(\underline{x}) &= 10 + 11x_1 - 8x_2 + x_3 \\u_2(\underline{x}) &= 50 - 10x_1 - 2x_2 - 8x_3\end{aligned}\tag{2-34}$$

In each law, six of the ten coefficients are very nearly zero (10^{-5} or smaller) and are ignored. The fact

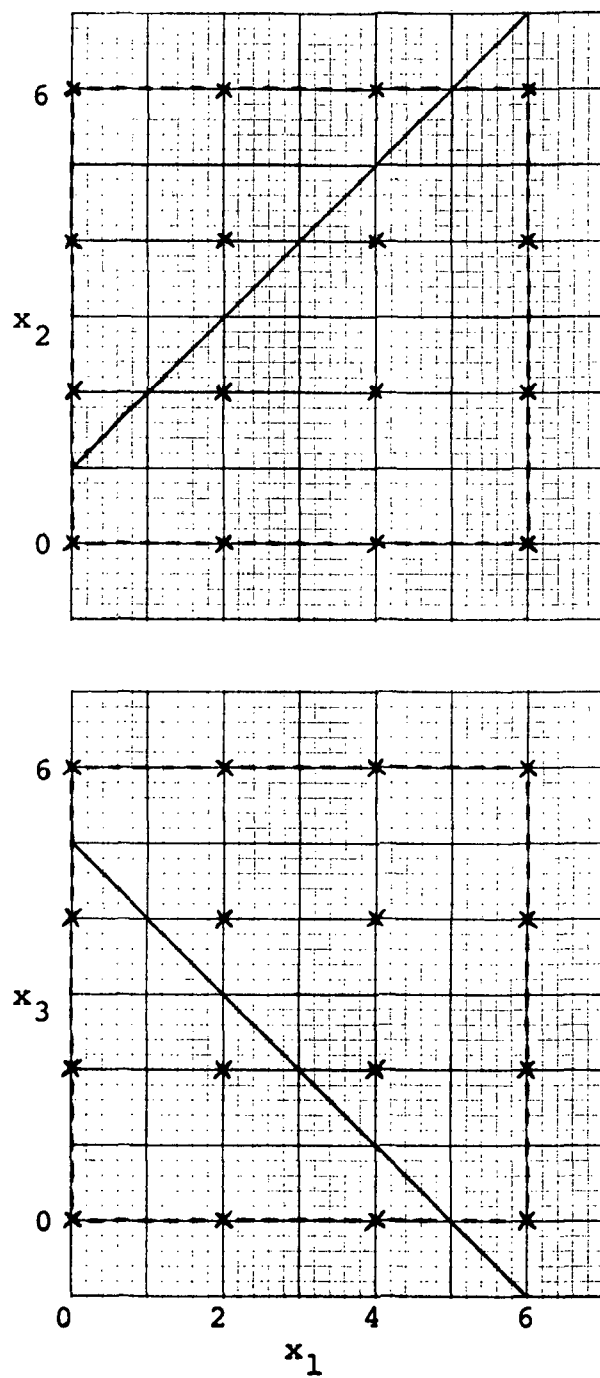


Fig. 2-2. Divergent Trajectory Specification for the Linear System Example

that these six are not exactly zero can be attributed to an inexact numerical fit procedure. Thus the control laws constitute simple linear feedback in this case.

The on-line controller operation is the same as outlined in the previous section except that step two, computation of the block number, is unnecessary since state space is not subdivided. Control update period is .05 seconds.

For the initial state $\underline{x} = (0,1,5)$ controller performance is easily checked against the analytic time solutions,

$$\begin{aligned}x_1^*(t) &= e^t - 1 \\x_2^*(t) &= e^t \\x_3^*(t) &= 6 - e^t \\u_1^*(t) &= 2e^t + 5 \\u_2^*(t) &= 12 - 4e^t\end{aligned}\tag{2-35}$$

A comparison between equations (2-34) and (2-35) demonstrates the validity of the control laws at points on the nominal trajectory. The digital system simulation of Fig. 2-3 verifies this. Equally important, Fig. 2-4 shows the plant behavior when started at a state which is not on the nominal trajectory. Recovery to the nominal is smooth and well behaved.

The second trajectory specification studied is shown in Fig. 2-5, in which the point $(0,0,3)$ is an equilibrium

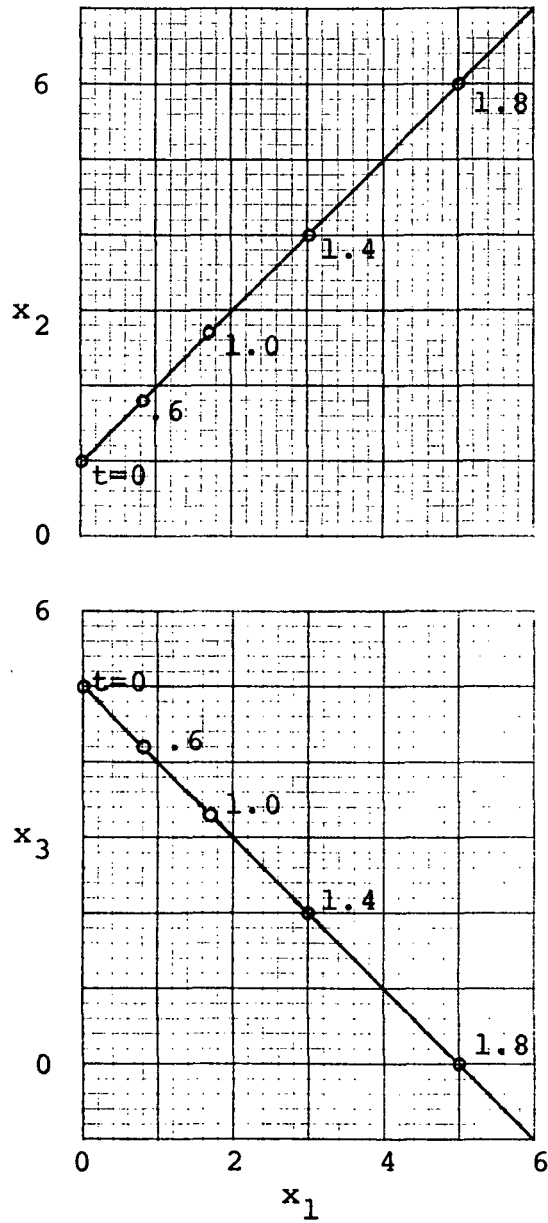


Fig. 2-3. Linear System Response with Divergent Trajectory Specification, $\underline{x}(0) = (0, 1, 5)$

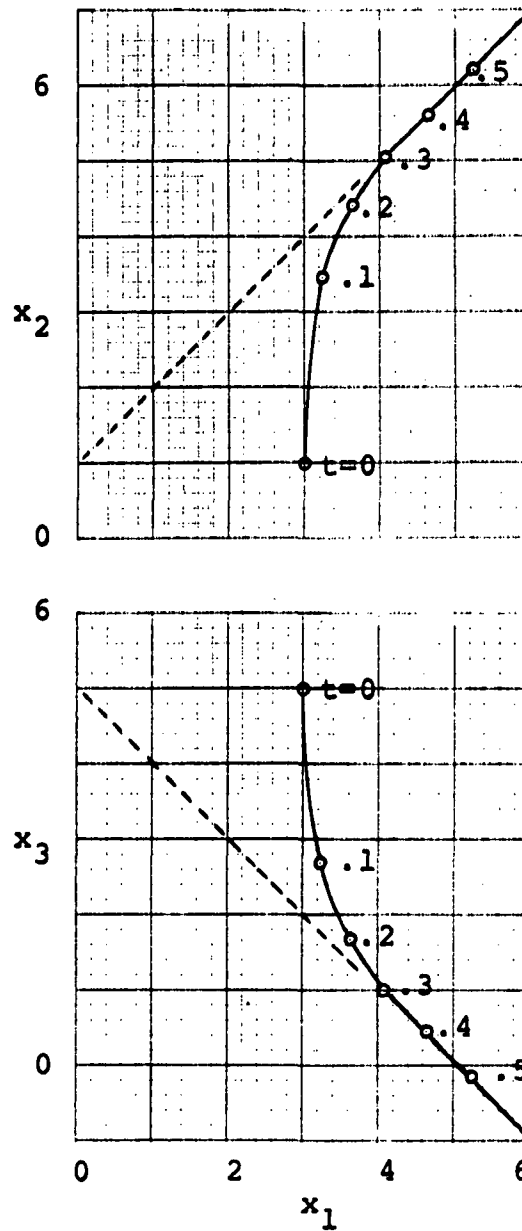


Fig. 2-4. Linear System Response with Divergent Trajectory Specification, $\underline{x}(0) = (3, 1, 5)$

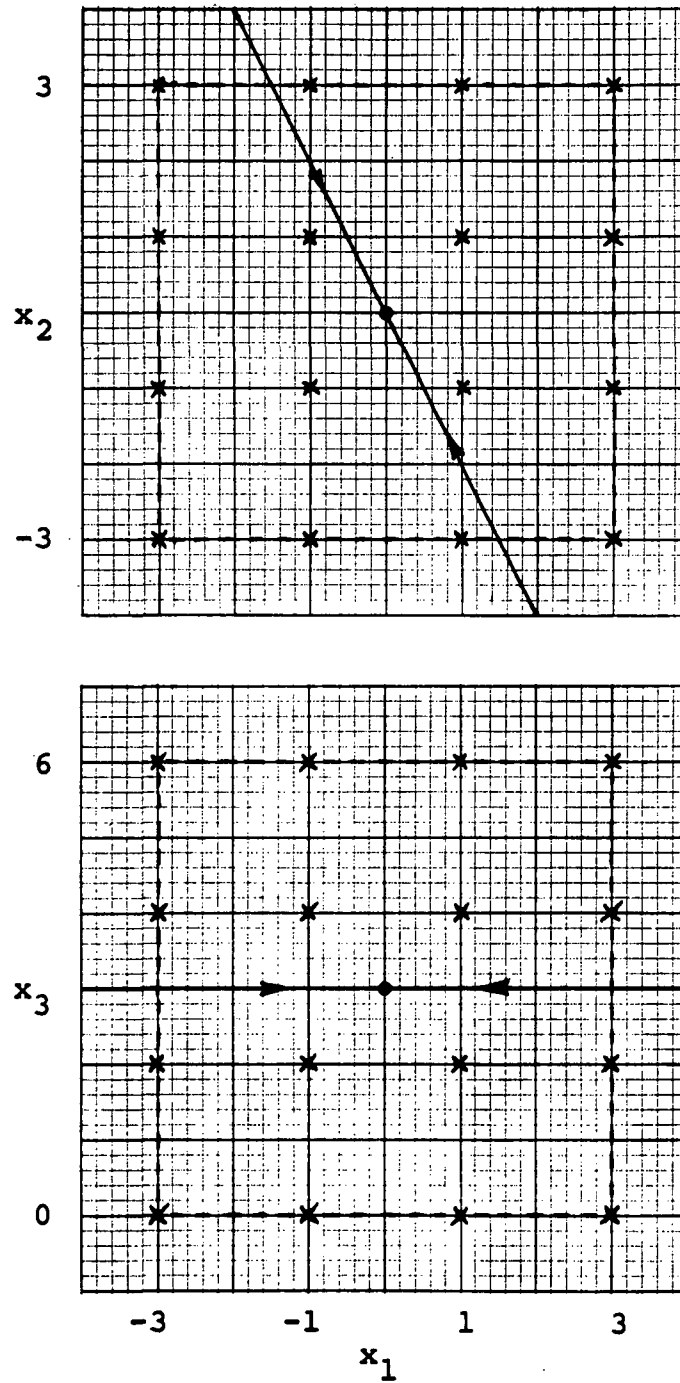


Fig. 2-5. Regulating Trajectory Specification for Linear System Example

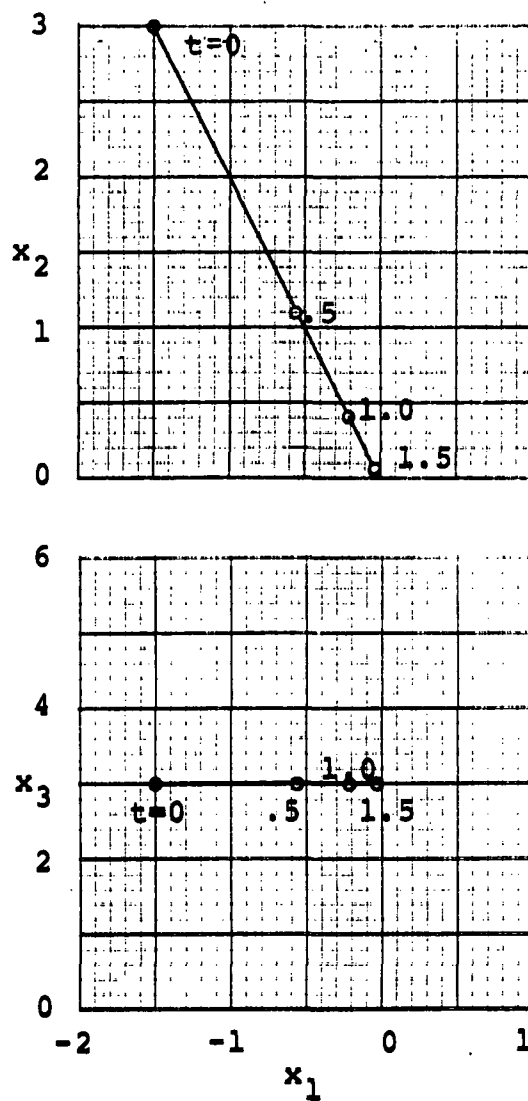


Fig. 2-6. Linear System Response with Regulating Trajectory Specification, $\underline{x}(0) = (-1.5, 3, 3)$

point. The controller for this case is designed in a manner completely similar to the previous case and the control laws are

$$\begin{aligned} u_1(\underline{x}) &= -19x_1 - 11x_2 + x_3 \\ u_2(\underline{x}) &= 30 - x_2 - 8x_3 \end{aligned} \tag{2-36}$$

When started on the nominal trajectory the plant remains there. When started off the nominal, the plant returns in a smooth, monotonic manner. No damped oscillation about the equilibrium point occurs even though the autonomous plant is very lightly damped. See Fig. 2-6.

It is interesting to note that the control laws computed for the single block are valid far beyond the block boundaries in both cases. This is not surprising since the plant and trajectory specifications are all linear. In the absence of control constraints, in fact, the laws should be globally valid.

Smith-Stenning Model Example

Smith and Stenning (1961) have proposed a highly simplified model of a nuclear rocket engine for use in conceptual studies. Several workers use this model to test the feasibility of various control schemes (Summary Report of Work to Date on the NERVA Digital Control System Design [NERVA] 1970, Kendrick 1972). Since the Smith-Stenning

model serves as a sort of reference standard in nuclear rocket engine control, it is chosen as an example here.

The equations given below are the same as the original set proposed by Smith and Stenning (1961), except that the various constants representing a NERVA class engine are substituted and the states are normalized to unity at a maximum power level typical for NERVA.

$$\begin{aligned}
 \dot{N} &= (65u_1 - .065\sqrt{T} + 19.5P/T - 65) N + 65C \\
 \dot{C} &= .1(N - C) \\
 \dot{T} &= 2(N - P\sqrt{T}) \\
 \dot{P} &= .4(Pu_2\sqrt{T} - P^2/\sqrt{T})
 \end{aligned}
 \tag{2-37}$$

where N = normalized neutron density (nuclear power level)

C = normalized precursor density

T = normalized temperature

P = normalized pressure

u_1 = normalized input reactivity

u_2 = normalized pressure ratio across turbine

The desired trajectory is expressed in terms of T , $\dot{T}$, and P ; therefore a simple transformation is necessary in order to create the state $\dot{T}$. The transformed set appears below with quantities as defined above and with B equal to normalized temperature rate.

$$\dot{T} = B$$

$$\begin{aligned} \dot{B} = & (B + 2P\sqrt{T}) [65u_1 - .065\sqrt{T} + 19.5P/T - 65] \\ & + 130C - P[B/\sqrt{T} + .8(u_2T - P)] \end{aligned} \quad (2-38)$$

$$\dot{P} = .4P\sqrt{T} (u_2 - P/T)$$

$$\dot{C} = .1(B/2 + P\sqrt{T} - C)$$

Because only two controls are available the trajectory can only be specified in two state space planes. In accordance with this limitation, the desired nominal trajectory, in terms of only three of the four states, is shown in Fig. 2-7. The reasons for choosing these three particular states are not presented here; they are similar to those cited in Chapter 4 in connection with the Simplified Nonlinear Model controller. It is sufficient here to say that this start-up trajectory takes the engine from Idle (low thrust) conditions at $T = .2$ and $P = .1$, to Design (full thrust) conditions at $T = .85$ and $P = .9$. The dotted lines in Fig. 2-7 indicate the values of T at the block boundaries. Applicable state space, that is the space immediately surrounding the nominal trajectory, is divided into eight blocks. The block in which the system resides is determined solely by the value of the independent state, T . Twenty-five points are chosen in each block where optimal controls are to be computed in the off-line process. Fig. 2-8 gives a two-dimensional indication of how these points are positioned.

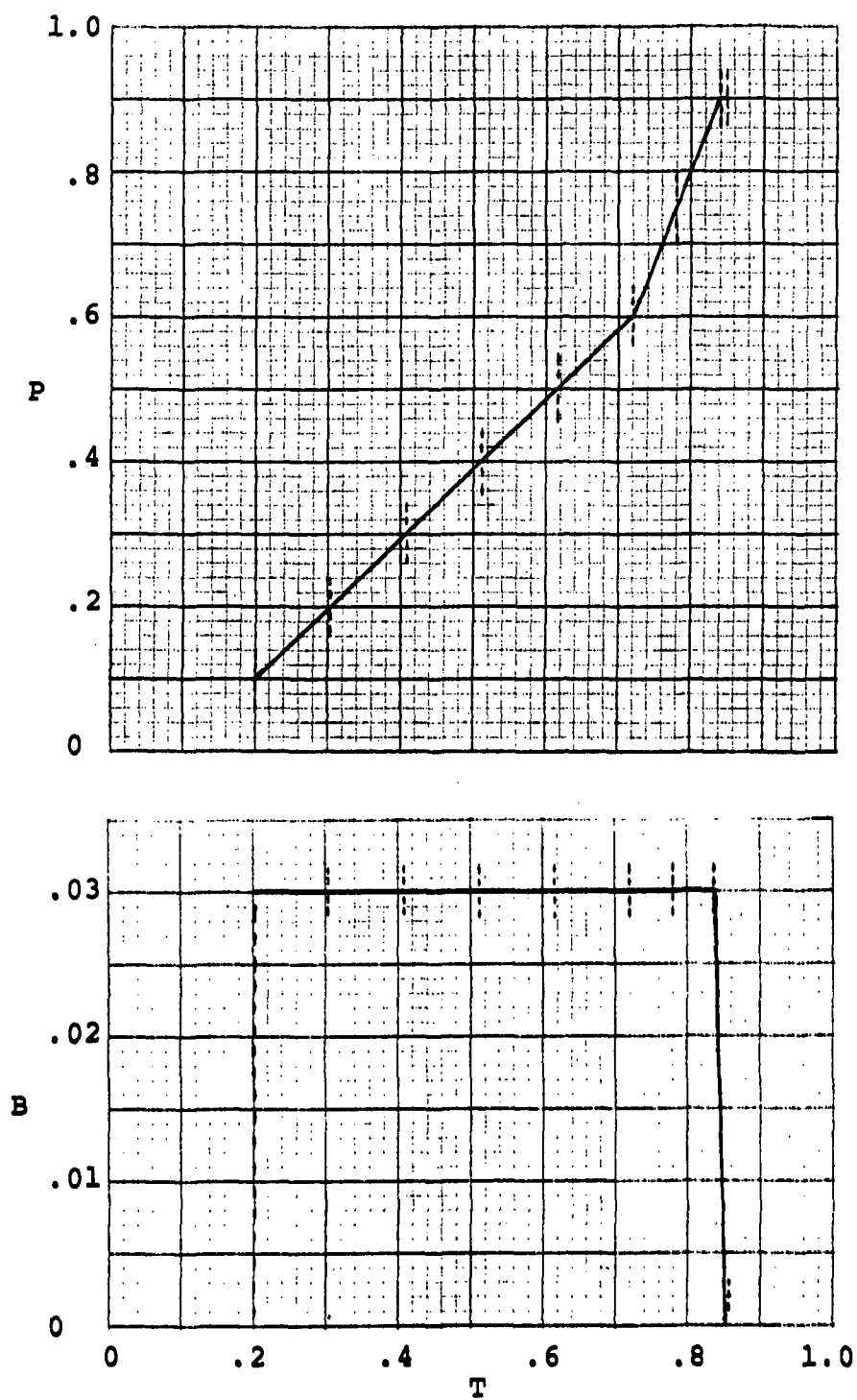


Fig. 2-7. Smith-Stenning Model Trajectory Specification

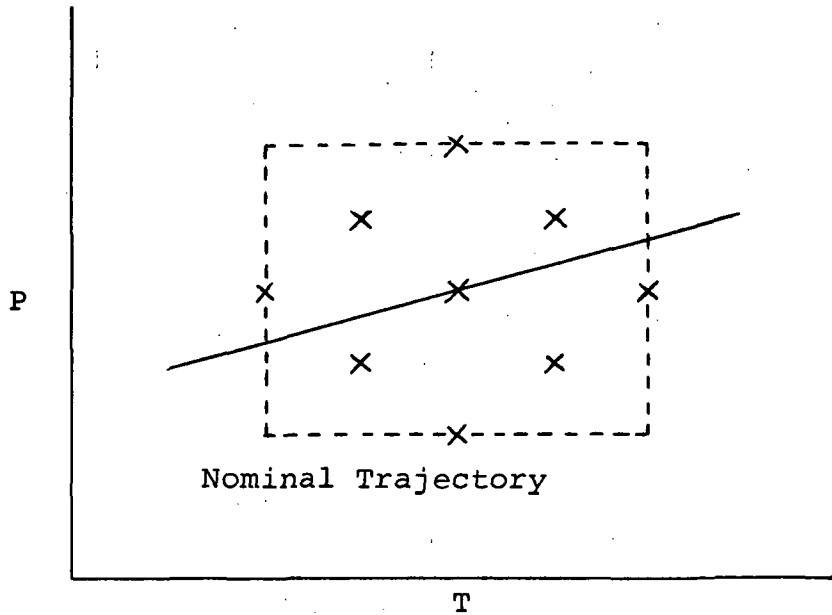


Fig. 2-8. Typical Block Geometry Used in Smith-Stenning Example Off-Line Optimizations

Since the control law is to be a function of all four states, the two approximate quadratic expressions describing u_1 and u_2 will, by equation (2-23), each require 15 coefficients. Thus, least squares fits are required over the 25 optimized points.

The performance index used in all optimizations is

$$J_r(\underline{u}) = \int_0^{dT} L[\underline{H}_r, \underline{x}(t)] dt + L[\underline{H}_r, \underline{x}(dT)] \quad (2-39)$$

$$dT = \max \left(7.44L[\underline{H}_r, \underline{x}], .075 \right)$$

where $L[\underline{H}_r, \underline{x}]$ is the orthogonal distance squared from point $\underline{x}$ to the line $\underline{H}_r = 0$ representing the nominal trajectory in

block r , and $\underline{x}(dT)$ is the system state as the result of control $\underline{u}$ applied for dT seconds.

Controller performance is evaluated both by digital simulations using the CDC 6400 Digital Computer and by real time analog simulations using the hybrid computer facility at Aerojet Nuclear Systems Co. Fig. 2-9 shows the startup trajectory with a control update period of .08 seconds which corresponds closely to the minimum value of dT used in off-line optimizations of .075 seconds. The transition between blocks caused by slight discontinuities in the control law are noticeable at several points in the $\dot{T}$ trace. These transitions are not evident in the T or P traces, however. The fast time constant associated with the nuclear power level can also be seen in the $\dot{T}$ trace as small variations between control updates, particularly for $t > 24.3$ with the system regulating at design conditions. Deviations from the specified (T,P) trajectory are extremely small.

The effects of changing the control update period are seen in the next figures. The update period is .02 second in Fig. 2-10 and .32 second in Fig. 2-11, below and above the nominal value by a factor of four. In Fig. 2-10 variations in $\dot{T}$ between updates have essentially disappeared as the system approaches continuous control. The very long update period in Fig. 2-11 shows degraded performance although the correct (T,P) trajectory is still followed quite closely. Comparing Figures 2-9 through 2-11, the system

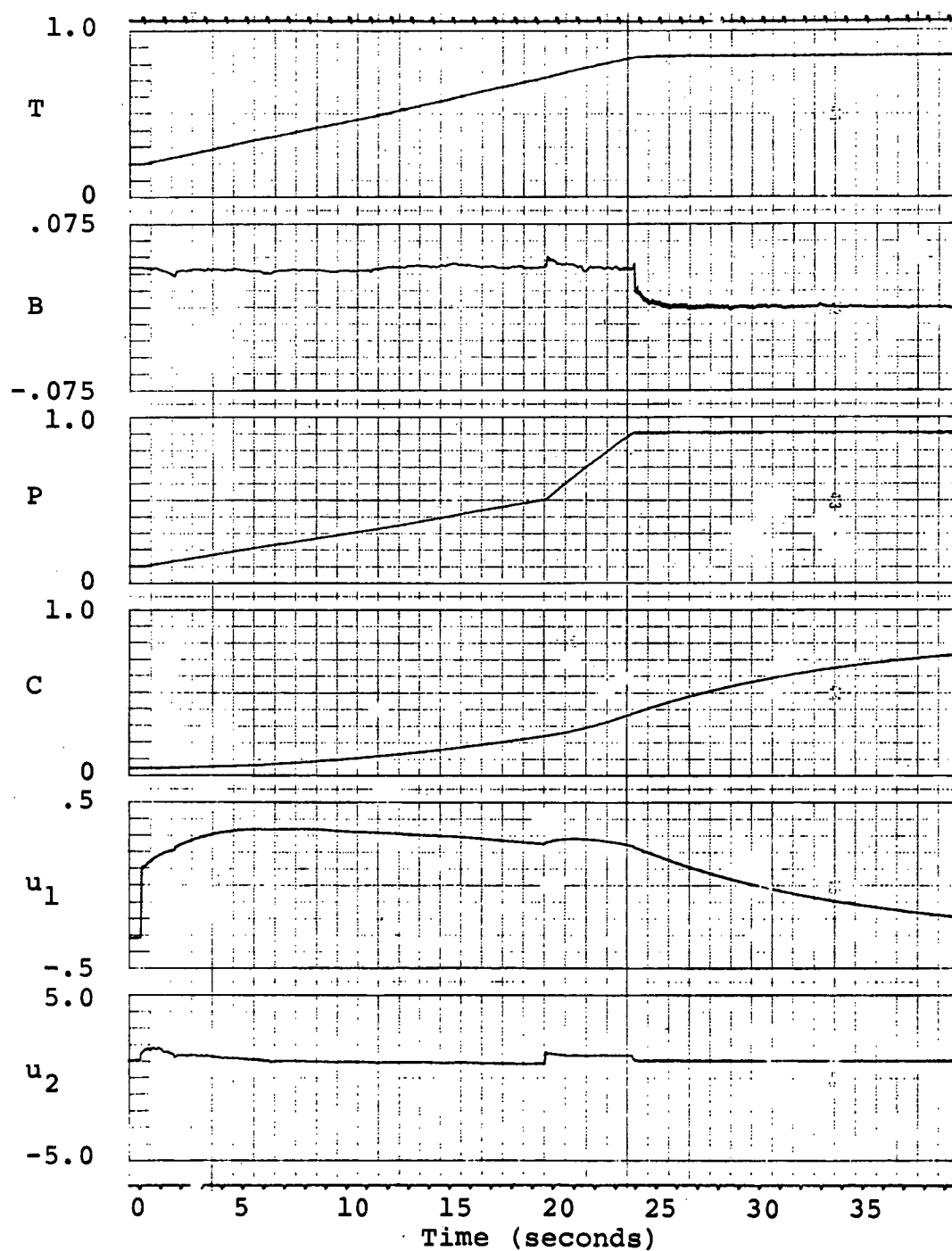


Fig. 2-9. Nominal Smith-Stenning Startup Time Response;
 $t_u = .08$

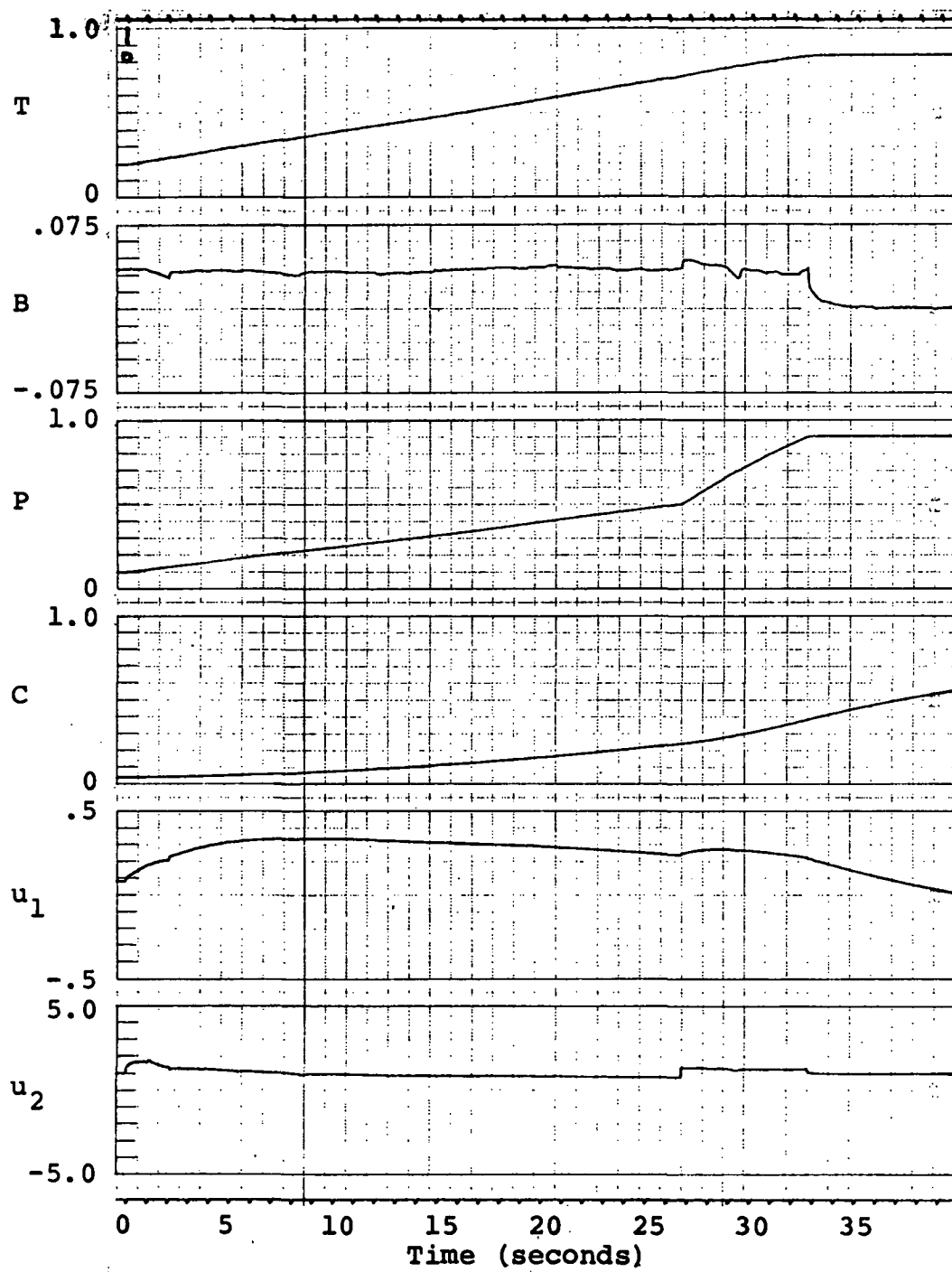


Fig. 2-10. Nominal Smith-Stenning Startup Time Response;
 $t_u = .02$

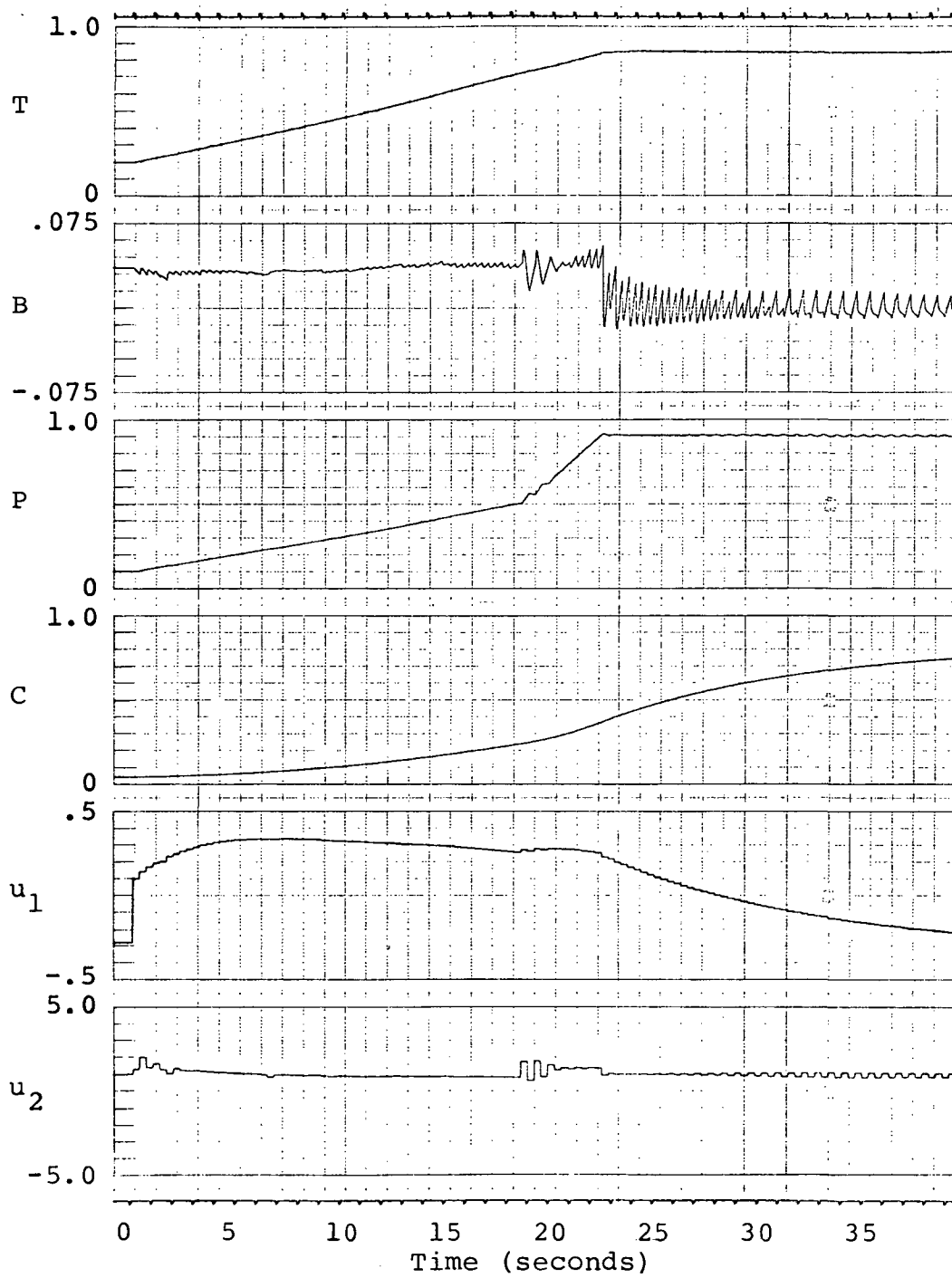


Fig. 2-11. Nominal Smith-Stenning Startup Time Response;
 $t_u = .32$

startup time appears to be different in each. This does not reflect a controller characteristic, but is due to the fact that the analog simulation was in the HOLD mode during update time even though the strip chart recorder ran continuously.

Figures 2-12 and 2-13 show system response to initial errors in pressure and precursor level of 150 and 50 percent respectively. "Error" in the initial precursor level is not really meaningful since this state is not included in the original trajectory specification. Nevertheless there exists a "nominal" response which corresponds to the nominal trajectories of the other three states and the controller is capable of returning the system to the nominal trajectory in spite of perturbations in precursor level.

Summary

The Precomputed State Dependent Control method has been described in this chapter. Several examples have demonstrated the ability of PSDC controllers to maintain linear and nonlinear multi-input systems along desired trajectories which are expressed explicitly as state functions.

Chapter 4 describes a particular application of PSDC: that of controlling the Simplified Nonlinear Model of NERVA. Because the SNM does not lend itself to direct application, however, a Control Model more suitable for off-line design use is first developed in Chapter 3.

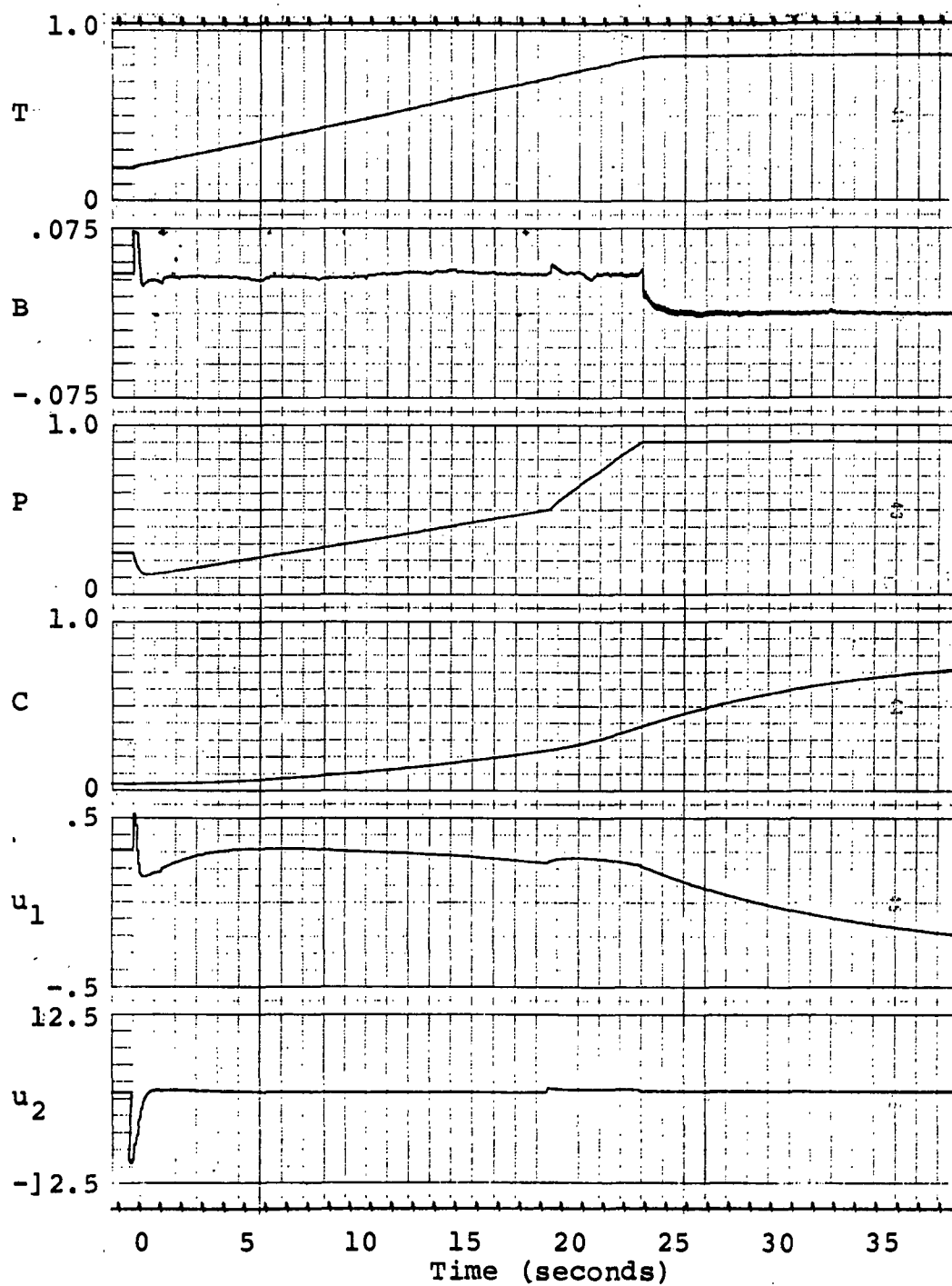


Fig. 2-12. Smith-Stenning Startup Time Response with +150% Initial Error in P; $t_u = .08$

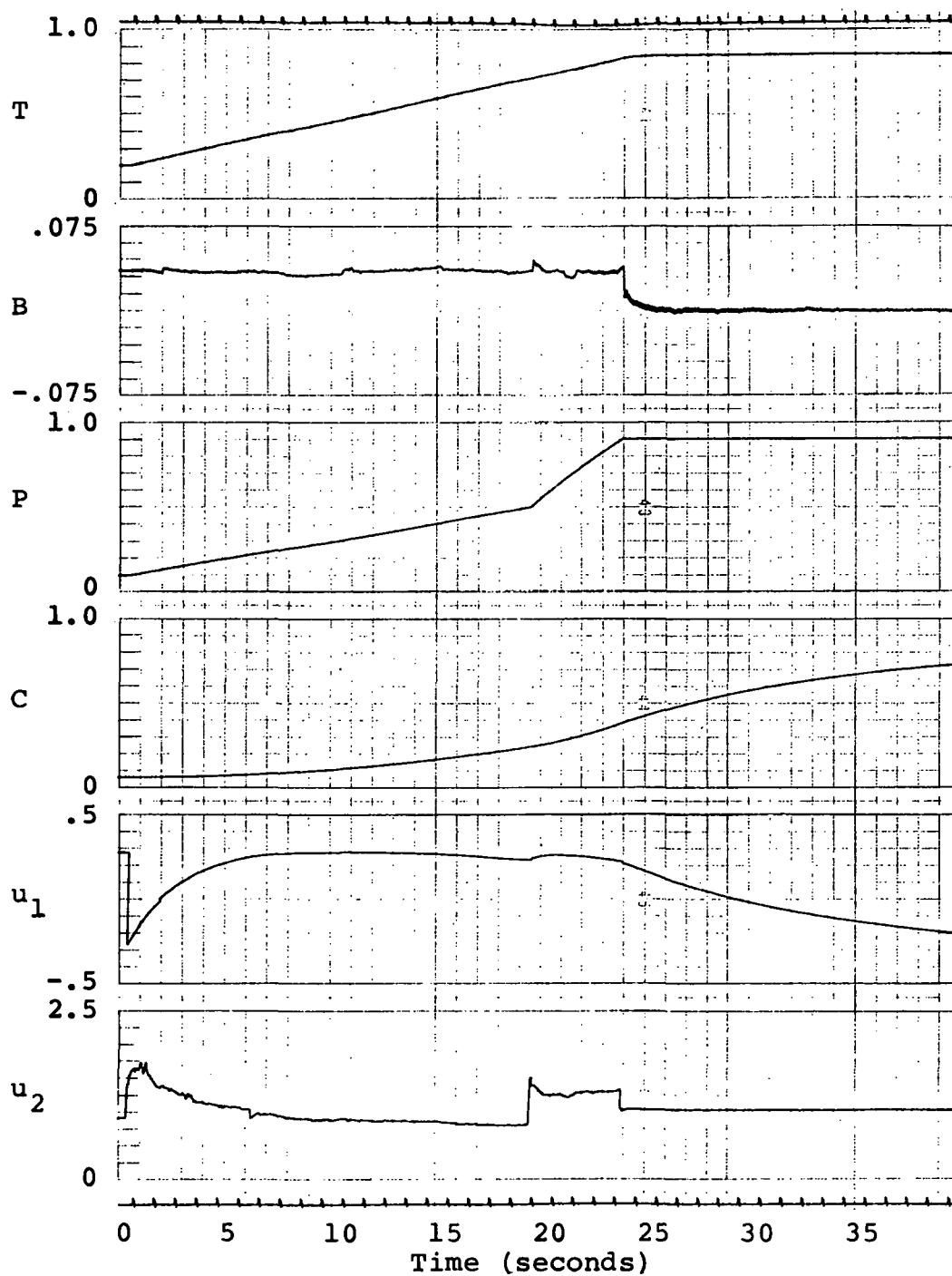


Fig. 2-13. Smith-Stenning Startup Time Response with +50% Initial Error in C; $t_u = .08$

CHAPTER 3

CONTROL MODEL DEVELOPMENT

The application of the PSDC method described in Chapter 2 presupposes that a suitable model of the system to be controlled exists. Two main considerations are involved in the question of suitability.

First there is the obvious problem of dimensionality. If a large number of states are required to adequately describe the dynamic behavior of a system, the off-line computational effort involved in the PSDC controller design may be prohibitive. (On-line requirements do not increase exponentially with system order as do off-line requirements, but are approximately proportional to the system order squared.)

Second, since the desired trajectory is to be expressed in terms of system states, those same states must exist in the system differential equation set. Thus a given model of the system to be controlled may not be usable in its initial form.

This chapter describes the development of a suitable control model which can be used to design a PSDC controller for the NERVA system. The evolution of the control model is traced from the most detailed model of NERVA presently available, down to one whose order is low enough to make the

PSDC method practicable and which contains the desired state vector.

The Common Analog Model

The Common Analog Model or CAM is the name given to a very detailed high order representation of the NERVA system. It has been developed and refined over several years by Aerojet and Westinghouse Corporations, and consists of 52 differential equations and numerous nonlinear algebraic equations and table functions.

The CAM is implemented by a large hybrid computer simulation at the Aerojet facility in Sacramento, California. Two COMCOR 5000 and two EAI 321R Analog Computers, a SIGMA 5 Digital Computer, and the necessary interface equipment comprise this real time simulation.

Although the CAM constitutes a valuable analysis tool it is too complex to provide a basis for controller design. It is for this reason that Kendrick (1972) developed a simplified model of NERVA.

The Simplified Nonlinear Model

In order to facilitate the application of his State Dependent State Variable Feedback control method to the NERVA system, Kendrick (1972) has developed a Simplified Nonlinear Model (SNM) which retains the important characteristics of the CAM. The SNM is developed on the basis of matching the static behavior and frequency response of the

linearized CAM at various power levels. Although the SNM is only eleventh order, Kendrick has shown it to be a good representation of the CAM. This is apparent both from direct static/dynamic comparisons, and from the fact that it was possible to control the CAM using a controller whose design was based on the SNM.

The SNM consists of the 11 differential and 11 implicit algebraic equations given below. Several of these are in a slightly different and more usable form than those in the original SNM. Also, several constants have been altered slightly to reflect recent refinements in the CAM.

$$\dot{T}_{15} = .02019W_n(T_{c2} - T_{15}) + 16.25S \quad (3-1)$$

$$\dot{T}_{c2} = .02671W_n(T_{c1} - T_{c2}) + 31.28S \quad (3-2)$$

$$\dot{T}_{c1} = -.01105W_n T_{c1} + 21.31S \quad (3-3)$$

$$\dot{S} = .0276C_1 + .235C_2 + 1.65C_3 - (1 - DKT) 660S \quad (3-4)$$

$$DKT = .55 W_{ss} + 3.25 \frac{P_{13} + P_{15}}{T_c} + .0244 \frac{P_2 + P_{11}}{T_7} - 7.7 \times 10^{-4} (T_c - 500) + DKD \quad (3-5)$$

$$T_c = .13(3.057T_{15} + 2T_{c2} + 3T_{c1}) \quad (3-6)$$

$$\dot{C}_1 = 181S - .0276C_1 \quad (3-7)$$

$$\dot{C}_2 = 369S - .235C_2 \quad (3-8)$$

$$\dot{C}_3 = 110S - 1.65C_3 \quad (3-9)$$

$$P_{15} = .076W_n\sqrt{T_{15}} \quad (3-10)$$

$$W_n = W_{bcv} + W_T \quad (3-11)$$

$$W_{bcv} = .04\theta_{bcv} \sqrt{\frac{P_{11}}{T_{11}}(P_{11} - P_{13})} \quad (3-12)$$

$$P_{13} = P_{15} \left(1 + \frac{33.3}{\sqrt{T_{15}}}\right) \quad (3-13)$$

$$\dot{W}_T = 10(W_{11} - W_{bcv} - W_T) \quad (3-14)$$

$$W_{11} = W_{no} + W_{ss} \quad (3-15)$$

$$W_{no} = 1.5 \sqrt{\frac{P_2 + P_{11}}{2T_7}}(P_2 - P_{11}) \quad (3-16)$$

$$W_{ss} = .0184\theta_{ssv} \sqrt{\frac{P_2 + P_{11}}{2T_7}}(P_2 - P_{13}) \quad (3-17)$$

$$P_{11} = .346W_T\sqrt{T_{11}} + .786P_{13} \quad (3-18)$$

$$P_2 = 3.434 \times 10^{-6}N^2 - 2.967 \times 10^{-4}W_{11}N + 30 \quad (3-19)$$

$$\begin{aligned} \dot{N} = 9.6 \times 10^5 \frac{W_T T_{11}}{N} \left[1 - \left(\frac{P_{13}}{P_{11}} \right)^{.25} \right] - 8.46 \times 10^{-5}N^2 \\ - .021W_{11}N \end{aligned} \quad (3-20)$$

$$\dot{T}_7 = -.011W_{11}(T_7 - 45) + 1.59S \quad (3-21)$$

$$\dot{T}_{11} = -.011W_{11}(T_{11} - 45) + 2.53S \quad (3-22)$$

where T_{15} = exit propellant temperature ($^{\circ}R$)
 T_{c2} = station 2 propellant temperature ($^{\circ}R$)
 T_{c1} = station 1 propellant temperature ($^{\circ}R$)
 S = nuclear power level (% full power)

DKT = total reactivity (\$)
 DKD = drum (input) reactivity (\$)
 T_c = average propellant temperature ($^{\circ}\text{R}$)
 C_1 = first precursor density
 C_2 = second precursor density
 C_3 = third precursor density
 P_{15} = exit pressure (lb/in^2)
 P_{13} = pressure at turbine exit (lb/in^2)
 P_{11} = pressure at turbine entrance (lb/in^2)
 P_2 = pressure at pump outlet (lb/in^2)
 W_n = hydrogen flow through core (lb/sec)
 W_{bcv} = flow through turbine by-pass valve (lb/sec)
 W_T = flow through turbine (lb/sec)
 W_{11} = flow through pump (lb/sec)
 W_{no} = flow through skirt (lb/sec)
 W_{ss} = flow through support structure (lb/sec)
 N = turbopump speed (rpm)
 T_7 = average temperature of hydrogen in skirt ($^{\circ}\text{R}$)
 T_{11} = temperature of hydrogen at turbine inlet ($^{\circ}\text{R}$)
 θ_{bcv} = turbine bypass valve position (% open)
 θ_{ssv} = support structure valve position = 13.1

Control Model Derivation

Theoretically, the PSDC method could be applied directly to the SNM. There are two practical reasons why this is not done. First, an eleventh order PSDC controller

design would require a vast amount of off-line computation. Second, as is pointed out in Chapter 4, the nominal trajectory specification is best made in terms of the quantities T_{15} , $\dot{T}_{15}$, and P_{15} . Of these three, only T_{15} exists as a state in the SNM.

This section contains the derivation of a Control Model or CM, beginning with the SNM, which overcomes both of the practical difficulties stated above. This derivation is accomplished in three steps. The first two steps concern certain approximations which reduce the model order from 11 to 4, and the third step is a transformation of variables to obtain the desired states.

Neutronics Approximation

The first step in the CM development involves approximations in the SNM neutronics. A subsystem of the SNM can be created by treating S as a pseudo-input and ignoring equations (3-4) through (3-9) in off-line computations. This is justified as follows.

Equation (3-4) contains an extremely fast time constant in S which can be neglected in the overall dynamics of the SNM. If the total reactivity, DKT , is changed suddenly due to a sudden change in input reactivity, DKD , the corresponding change in S occurs in a few milliseconds. Of course DKT is affected by other quantities as equation (3-5) shows, but these terms change very slowly. Also, S is a

function of the three precursors as well as DKT. But again, changes in precursor levels occur very slowly. Thus, except at points in time where DKT is discontinuous, S changes very slowly and $\dot{S} \approx 0$. Equation (3-4) then reduces to

$$S \approx \frac{.0276C_1 + .235C_2 + 1.65C_3}{660(1 - DKT)} \quad (3-23)$$

and the assumption is that the on-line control update rate is chosen so that variations in S between updates can be neglected. On-line, S and DKT are measured and the precursor sum is found using (3-23). Knowing the desired value of S from the CM controller, (3-23) is then used again to find the required change in DKT and hence the change in the actual input DKT. This process is explained more fully in Chapter 4.

In this manner the four states S , C_1 , C_2 , and C_3 are removed for the purpose of off-line controller design computation. This first step in the evolution of the CM is shown in Fig. 3-1.

Temperature Approximations

Consider the three temperature equations (3-3), (3-21), and (3-22). If W_{11} and W_n are assumed to be constant for the moment, these are all first order linear differential equations forced by S . If T_{15} is forced along a time ramp, which is the case along the nominal trajectory to be specified, then the other temperature functions will be

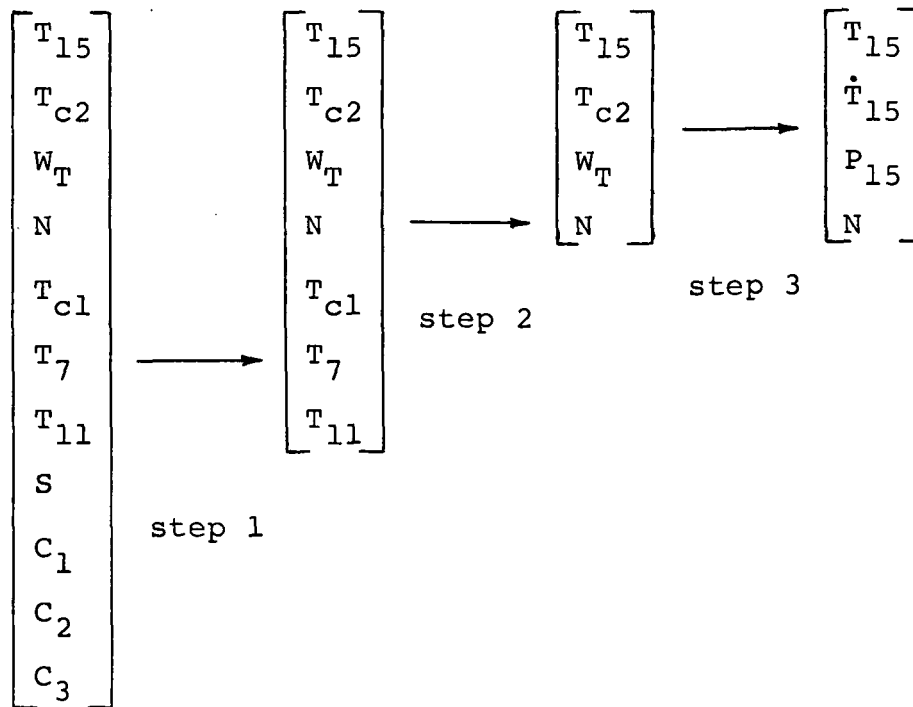


Fig. 3-1. CM Controller State Vector Evolution

ramps which lead or lag T_{15} according to their faster or slower time constants. Therefore, except for a short transient period at the beginning of the ramp or at points where the ramp slope changes, the three temperatures may be approximated as functions of T_{15} and $\dot{T}_{15}$. For the slow time constants encountered in these temperature equations, the approximation

$$W_{11} \approx W_n \quad (3-24)$$

is a good one, and by including W_n in the arguments of these three approximate temperature equations the assumption of constant W_n and W_{11} can be lifted. The approximate expressions for T_{c1} , T_7 , and T_{11} are given in equations (3-31),

(3-32), and (3-33) respectively and the details of the derivations are found in Appendix A.

Since T_{c1} , T_7 , and T_{11} are now approximated as functions of other variables they are no longer considered "states" in the equation set. Step two is thus completed, and is depicted in Fig. 3-1.

Transformation of Variables

The last step in the CM development involves a variable transformation of the remaining states T_{15} , T_{c2} , W_T , and N , to obtain the desired state vector, T_{15} , $\dot{T}_{15}$, P_{15} , and N . In order to maintain text continuity, the details of this transformation are given in Appendix A, and only the results are given in the following subsection.

Control Model Summary

The three development steps presented above are summarized here in a complete mathematical statement of the Control Model. The CM consists of the four differential and ten algebraic equations (none of which are implicit) given below.

$$\dot{T}_{15} = B \quad (3-25)$$

$$\begin{aligned} \dot{B} = & .02019W_n [.02671W_n (T_{c1} - T_{15}) - 2.323B + 52.78S] \\ & + \frac{W_{nd}}{W_n} (B - 16.25S) \end{aligned} \quad (3-26)$$

$$\dot{P}_{15} = .076 \frac{W_n B}{2\sqrt{T_{15}}} + W_{nd}\sqrt{T_{15}} \quad (3-27)$$

$$\begin{aligned} \dot{N} = 9.6 \times 10^5 \frac{W_{T11}}{N} \left[1 - \left(\frac{P_{13}}{P_{11}} \right)^{.25} \right] - 8.46 \times 10^{-5} N^2 \\ + .021 W_{11} N \end{aligned} \quad (3-28)$$

$$P_{13} = P_{15} \left(1 + \frac{33.3}{\sqrt{T_{15}}} \right) \quad (3-29)$$

$$W_n = \frac{P_{15}}{.076\sqrt{T_{15}}} \quad (3-30)$$

$$T_{c1} = .4939T_{15} + \frac{16.25B}{W_n} \quad (3-31)$$

$$T_7 = .03702T_{15} + \frac{1.223B}{W_n} + 45 \quad (3-32)$$

$$T_{11} = .05891T_{15} + \frac{1.946B}{W_n} + 45 \quad (3-33)$$

$$W_{bcv} = f_1(W_n, T_{11}, P_{13}, \theta_{bcv}) \quad (3-34)$$

$$W_T = W_n - W_{bcv} \quad (3-35)$$

$$P_{11} = .346W_T\sqrt{T_{11}} + .786P_{13} \quad (3-36)$$

$$W_{11} = f_2(P_{11}, N, T_7) \quad (3-37)$$

$$W_{nd} = f_3(W_{bcv}, W_T, W_n, W_{11}, P_{11}, P_{13}, T_{11}, T_{15}, B, \theta_{bcv}) \quad (3-38)$$

where f_1 , f_2 , and f_3 are given in Appendix A. Except for the fact that the equations would be cumbersome, (3-29) through (3-38) could be substituted into (3-26) through (3-28) and the entire set written in the classical form of

equation (2-8) as

$$\begin{aligned}
 \dot{T}_{15} &= g_1(B) \\
 \dot{B} &= g_2(T_{15}, B, P_{15}, N, S, \theta_{bcv}) \\
 \dot{P}_{15} &= g_3(T_{15}, B, P_{15}, N, \theta_{bcv}) \\
 \dot{N} &= g_4(T_{15}, B, P_{15}, N, \theta_{bcv})
 \end{aligned}
 \tag{3-39}$$

Summary

The Control Model which is to be used in the design of a PSDC controller for the SNM, is developed in this chapter. The CM is based upon the SNM and its derivation consists of the elimination of certain differential equations and a transformation of the remaining ones.

Four of the SNM differential equations are eliminated by considering S , the nuclear power level, to be a pseudo-input and ignoring the neutronics equations. Thus the CM is not, strictly speaking, a complete representation of the SNM. However, the neutronics are included in the actual on-line control of the SNM as is explained in the next chapter. Next, three other differential equations are eliminated by developing approximate relationships between three system temperatures and the exit propellant temperature and other remaining states. Finally, a transformation of variables is performed to create the states of direct interest at the expense of states of lesser importance.

In light of the approximations made in the derivation of this Control Model, the reader may question the extent to which it is a valid representation of the SNM. The ultimate test of the CM is an examination of the SNM control system behavior, because the SNM controller design is to be based on the CM. Chapter 4 describes the design of PSDC controllers for both the CM and SNM control systems, and Chapter 5 describes the performance of these two control systems. It is shown in these chapters that differences between the two models are primarily due to the neutronics approximations and, as a result, slight modifications in the SNM controller design procedure are necessary.

CHAPTER 4

DESIGN OF THE CONTROL MODEL AND SIMPLIFIED NONLINEAR MODEL CONTROLLERS

This chapter describes the design of two controllers using the PSDC method. First, a CM controller is designed to drive the Control Model along a specific state space trajectory. Second, an SNM controller is designed to drive the Simplified Nonlinear Model along the same trajectory. Because the CM has only four states and because the desired trajectory can be specified in terms of these states, application of the PSDC method is possible, and the CM controller design is accomplished in a straightforward manner.

Direct application of the PSDC method to the Simplified Nonlinear Model is difficult because of the large number of states involved. Thus, the SNM controller design is based completely on the CM; it is for this reason that the CM was developed. The hope, of course, is that a controller whose design is based on the CM is capable of controlling the higher order SNM in an acceptable manner. Experiments with the SNM control system show that this is the case.

First, the various aspects of the CM controller design are presented, beginning with a detailed examination of the nominal trajectory specification to be used. Both

off-line design and on-line operational considerations are included.

Control Model Controller Off-Line Design

Application of the PSDC method cannot be described as algorithmic. The nominal trajectory to be specified depends upon such things as system efficiency and physical constraints. The division of state space into blocks, and the geometry of these blocks, must be specified. The performance indices used in the off-line optimizations and the analytic form of the control law to be used must be chosen. Generally, some experimentation is required before these decisions can be made.

This section summarizes the off-line portion of the CM controller design. Each of the aspects mentioned are discussed and corresponding choices documented. Though the individual choices may not be optimal in any particular sense, they do result in a controller having good performance.

Nominal Trajectory Specification

Fig. 4-1 shows time plots of T_{15} and P_{15} for a normal NERVA engine startup. Beginning at 1200°R and 60 psi which is designated Idle Point, T_{15} is increased at a constant rate of about 12 psi/sec to 300 psi. This condition is designated Throttle Point. At this time, T_{15} is held

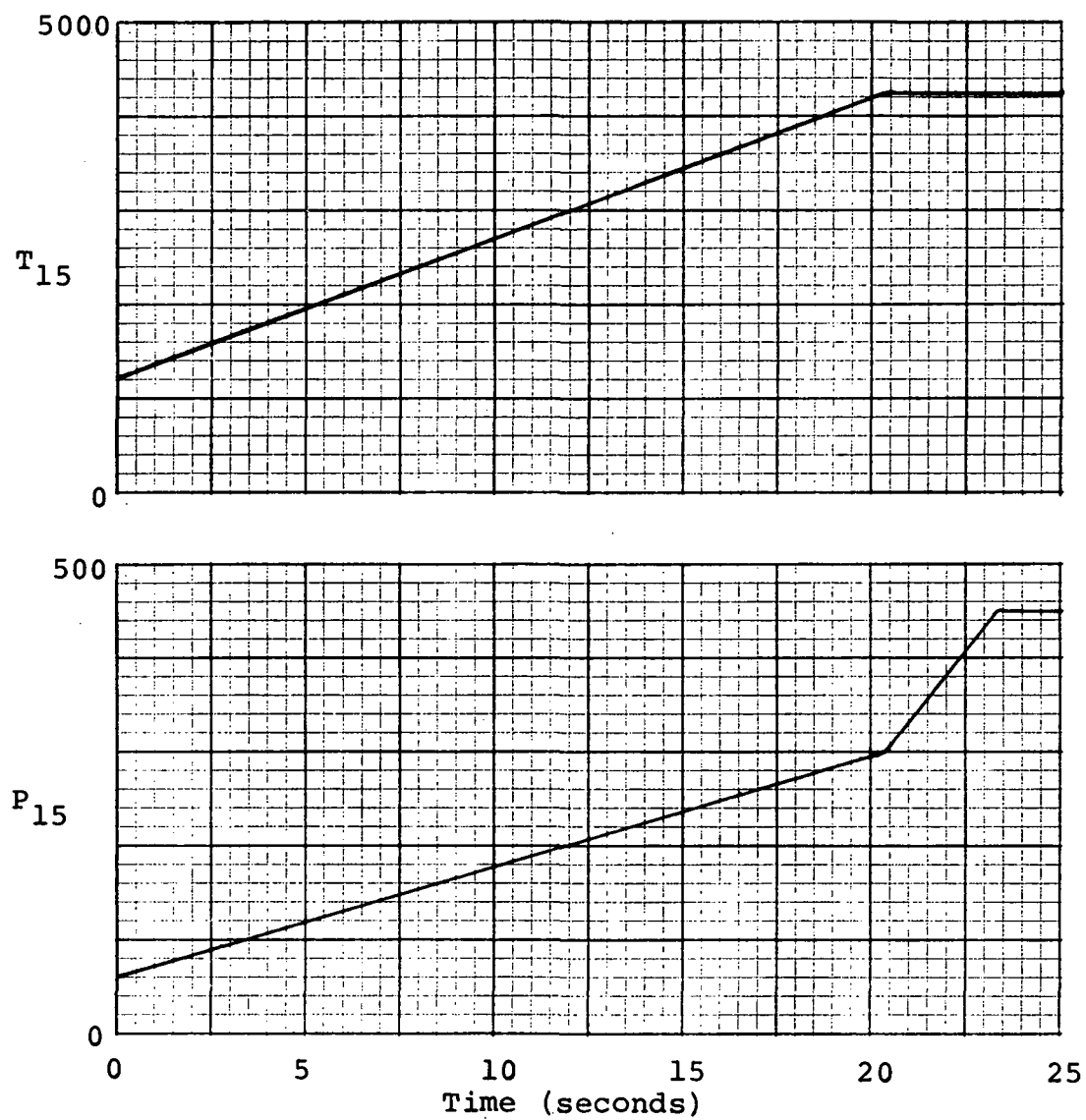


Fig. 4-1. Normal NERVA Startup Temperature and Pressure Time Profiles

constant and P_{15} increases at 50 psi/sec to its maximum value of 450 psi. Ideal behavior in returning from Design to Idle conditions (shutdown) is an image of these plots, P_{15} returning to 300 psi at -50 psi/sec, then P_{15} and T_{15} decreasing linearly to Idle.

These time plots can be partially expressed as a trajectory in the (T_{15}, P_{15}) plane as shown in Fig. 4-2. The time rates shown in the time profiles can only be expressed in terms of $\dot{T}_{15}$ and/or $\dot{P}_{15}$, however. Specifically, from Idle to Throttle Point, either one of the sets $(T_{15}, \dot{T}_{15}, P_{15})$ or $(T_{15}, P_{15}, \dot{P}_{15})$ are necessary for a complete trajectory specification. From Throttle to Design Point only the latter set is acceptable since $\dot{T}_{15} = 0$ along this segment of the trajectory.

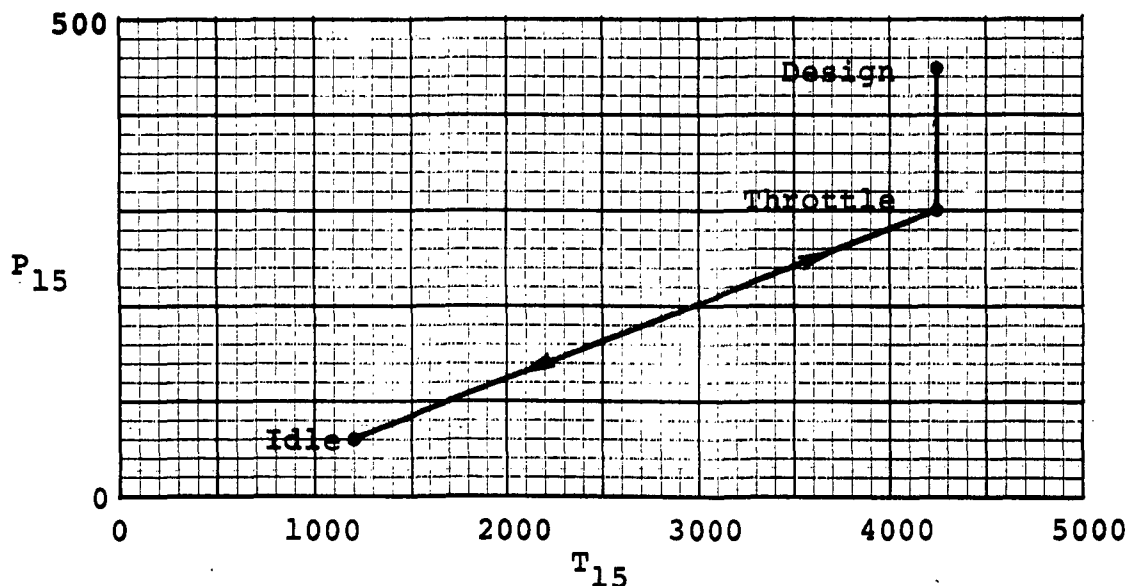


Fig. 4-2. Partial State Space Representation of NERVA Engine Startup

The development of the Control Model in Chapter 3 includes the creation of the state $\dot{T}_{15} = B$. The creation of the state $\dot{P}_{15}$ presented a number of analytic difficulties. With the state variable set thus limited, the trajectory of Fig. 4-2 is not acceptable. What is desired from a computational point of view, is a slightly different trajectory specification which retains the advantages of the original and yet which can be specified in terms of the set (T_{15}, B, P_{15}) . Such a trajectory is shown in Fig. 4-3.

The normal specification of Figures 4-1 and 4-2 is a compromise between engine efficiency and a consideration of the temperature and pressure rate constraints imposed by the physical limitations of the engine structure. The modified trajectory specification of Fig. 4-3 is very similar as regards temperature and pressure time rates. It is interesting to compare the relative engine startup efficiencies resulting from the two trajectories.

$$\begin{aligned} \text{Relative Efficiency} &= \frac{\text{Total Impulse}}{\text{Coolant Consumption}} \\ &= \frac{\int_0^T P_{15}(t) dt}{\int_0^T \frac{P_{15}(t)}{\sqrt{T_{15}(t)}} dt} \end{aligned} \quad (4-1)$$

where T is the time required for the slowest of the two trajectories to arrive at Design Point conditions. Precisely

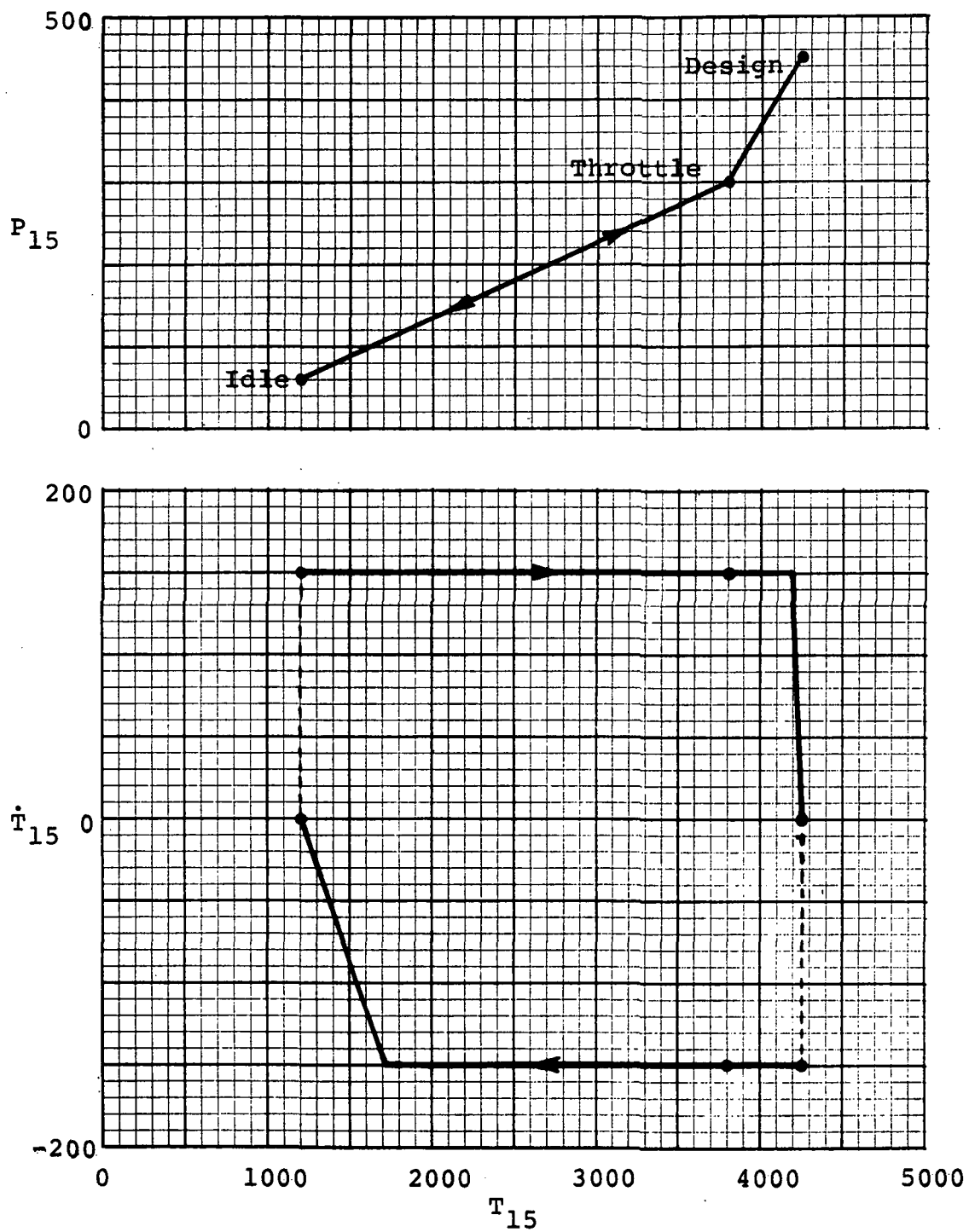


Fig. 4-3. Modified Trajectory Specification for CM Controller

along the two trajectories, P_{15} and T_{15} behave as follows.

For the normal trajectory,

$$\left. \begin{aligned} P_{15}(t) &= 11.803t + 60 \\ T_{15}(t) &= 150t + 1200 \end{aligned} \right\} \quad 0 \leq t \leq 20.333 \quad (4-2)$$

$$\left. \begin{aligned} P_{15}(t) &= 50(t - 20.333) + 300 \\ T_{15}(t) &= 4250 \end{aligned} \right\} \quad 20.333 \leq t \leq 23.333 \quad (4-3)$$

For the modified trajectory,

$$\left. \begin{aligned} P_{15}(t) &= 13.846t + 60 \\ T_{15}(t) &= 150t + 1200 \end{aligned} \right\} \quad 0 \leq t \leq 17.333 \quad (4-4)$$

$$\left. \begin{aligned} P_{15}(t) &= 50(t - 17.333) + 300 \\ T_{15}(t) &= 150t + 1200 \end{aligned} \right\} \quad 17.333 \leq t \leq 20 \quad (4-5)$$

$$\left. \begin{aligned} P_{15}(t) &= 450 - 16.667e^{-3(t - 20)} \\ T_{15}(t) &= 4250 - 50e^{-3(t - 20)} \end{aligned} \right\} \quad 20 \leq t \quad (4-6)$$

Assuming the exponential has disappeared in, say, three time constants, the modified trajectory causes the system to arrive at Design Point in 21 seconds or slightly faster than the normal trajectory; so let $T = 23.333$ seconds.

Using the above time functions, equation (4-1) gives relative efficiencies of 55.9 for the normal and 56.4 for the modified trajectory, actually an improvement of about 0.9 percent. The use of the modified specification is thus

justified, suffering no loss of efficiency and yet allowing use of the more convenient state variable set, (T_{15}, B, P_{15}) .

The modified trajectory specification is used throughout the remainder of this study, and is defined by the following equations. For startup,

$$\left. \begin{array}{l} B = 150 \\ P_{15} = .092308T_{15} - 50.769 \end{array} \right\} \quad 1200 \leq T_{15} \leq 3800 \quad (4-7)$$

$$\left. \begin{array}{l} B = 150 \\ P_{15} = .33333T_{15} - 966.67 \end{array} \right\} \quad 3800 \leq T_{15} \leq 4200 \quad (4-8)$$

$$\left. \begin{array}{l} B = -3T_{15} + 12750 \\ P_{15} = .33333T_{15} - 966.67 \end{array} \right\} \quad 4200 \leq T_{15} \leq 4250 \quad (4-9)$$

and for shutdown,

$$\left. \begin{array}{l} B = -150 \\ P_{15} = .33333T_{15} - 966.67 \end{array} \right\} \quad 3800 \leq T_{15} \leq 4250 \quad (4-10)$$

$$\left. \begin{array}{l} B = -150 \\ P_{15} = .092308T_{15} - 50.769 \end{array} \right\} \quad 1720 \leq T_{15} \leq 3800 \quad (4-11)$$

$$\left. \begin{array}{l} B = -.24769T_{15} + 297.23 \\ P_{15} = .092308T_{15} - 50.769 \end{array} \right\} \quad 1200 \leq T_{15} \leq 1720 \quad (4-12)$$

Division of State Space

Once a control law form is chosen, the off-line computational requirement and the on-line efficiency of a PSDC

controller is largely dependent upon the total number of blocks in which applicable state space is to be divided. If state space is divided too finely, the correspondingly large number of blocks increases the computational effort required. If divided too coarsely, the control law form may not adequately describe the optimal control law. A compromise is therefore necessary and in general the best division is found empirically.

The control law form chosen for both the CM and SNM controllers is the standard hyperellipsoidal form used in the Smith-Stenning example of Chapter 2. The nominal trajectory specification used in the example is quite similar to that used in the present application. Thus the experience gained by working with the Smith-Stenning model provides the means of dividing the CM state space in a reasonable way.

Table 4-1 shows the division of CM state space. The arrows indicate whether T_{15} is increasing or decreasing. Blocks 1 through 8 correspond to engine startup, and 9 through 15 to engine shutdown. Hence there are actually two controllers, and on-line provision must be made to switch from one mode to the other as desired. (See Fig. 4-7.)

In either mode the block in which the system resides is determined solely by the value of T_{15} . This greatly simplifies on-line control signal computations. Notice that the temperature range covered by each block varies according

Table 4-1. CM and SNM State Space Block Boundaries

Block No.	T_{15} Range	ΔT
1	1200 $\rightarrow$ 1720	0.5
2	1720 $\rightarrow$ 2240	0.5
3	2240 $\rightarrow$ 2760	0.4
4	2760 $\rightarrow$ 3280	0.4
5	3280 $\rightarrow$ 3800	0.3
6	3800 $\rightarrow$ 4000	0.3
7	4000 $\rightarrow$ 4200	0.3
8	4200 $\rightarrow$ 4250	0.3
9	4025 $\leftarrow$ 4250	0.3
10	3800 $\leftarrow$ 4025	0.3
11	3280 $\leftarrow$ 3800	0.3
12	2760 $\leftarrow$ 3280	0.4
13	2240 $\leftarrow$ 2760	0.4
14	1720 $\leftarrow$ 2240	0.5
15	1200 $\leftarrow$ 1720	0.5

to the corresponding range of other states along the nominal trajectory in that temperature region.

Blocks 8 and 15 are unique. In each of these blocks the B trajectory specification passes through zero and hence defines an equilibrium point. Block 8 tends to maintain the system at Design Point and block 15 at Idle Point.

Finally, a comparison between Table 4-1 and the nominal trajectory specification given in Fig. 4-3 shows that all discontinuities and corners in the trajectory specification occur at block boundaries. This simplifies the trajectory description in any single block and hence simplifies the off-line computations.

Block Geometry

The optimization point placement used in the 15 applicable state space blocks is chosen to obtain a valid control law over as wide a region as possible while retaining a good fit to the optimization data.

The question of how wide a "ribbon" of state space to consider surrounding the nominal trajectory, is an important one. The advantage of keeping the ribbon narrow is that the control law is more accurately approximated near the nominal trajectory. On the other hand if at least a few points quite far from the nominal trajectory are not included in the optimization process, there is no real assurance that the system could recover from such states. This is not

to say that the control law is valid only to the furthest point in the original optimization. Experimental results show that the CM and SNM control systems are capable of recovering smoothly from points several times the ribbon width from the nominal trajectory, indicating the validity of the control law far beyond the block boundaries.

As an indication of the ribbon width and point placement pattern used in the CM controller design, Table 4-2 gives the coordinates of the 25 optimization points chosen for block 4. When plotted in the three state space planes, the ribbon width is seen to be 100°R/sec , 40 psi, and 2000 rpm along the B, P_{15} , and N axes respectively. This width is used in all blocks of state space. Ribbon width along the T_{15} axis is not defined since T_{15} is the independent state in the trajectory specification.

The selection of the N-coordinates is of interest. Though the nominal trajectory specification excludes N, a "nominal" trajectory for N corresponding to this specification exists. It is extremely difficult to solve for this "nominal" analytically; instead it is approximated by static model data at various values of T_{15} and P_{15} . It is not important to obtain optimal control data at points exactly on this "nominal" N trajectory, but only that the region containing it is included in the data.

Table 4-2. Optimization Point Coordinates for Block 4,
CM and SNM Controllers

Point	T ₁₅	B	P ₁₅	N
1	2760	150	204	16189
2	3020	150	228	17090
3	3280	150	252	17941
4	3020	100	228	17090
5	3020	200	228	17090
6	3020	150	208	17090
7	3020	150	248	17090
8	3020	150	228	16090
9	3020	150	228	18090
10	2890	125	206	16147
11	2890	125	206	17147
12	2890	125	226	16147
13	2890	125	226	17147
14	2890	175	206	16147
15	2890	175	206	17147
16	2890	175	226	16147
17	2890	175	226	17147
18	3150	125	230	17022
19	3150	125	230	18022
20	3150	125	250	17022
21	3150	125	250	18022
22	3150	175	230	17022
23	3150	175	230	18022
24	3150	175	250	17022
25	3150	175	250	18022

Performance Index

The performance index used in all optimizations in the CM controller design is that of equation (2-19) which is repeated here for convenience.

$$I_r(\underline{x}) = \min_{\underline{u}} \left(L[\underline{H}_r, \underline{x}(dT)] \right) \quad (4-13)$$

The cost functional $L[\underline{H}_r, \underline{x}]$ is the euclidean error vector norm given by equation (2-21). This index results in controls which cause the system to return to some point on the nominal trajectory in exactly dT seconds. Indices described by equations (2-17) and (2-18) were investigated, but optimizations using these indices were more time consuming and did not result in better control laws. Smaller values of dT in the cost functional naturally reduce the optimization time and are desirable for this reason. The disadvantage is that as dT is reduced, the necessary control effort to return the system to the nominal trajectory from any given point in dT seconds is increased. The result is twofold. First, the resulting control law may demand controls which violate control constraints even when relatively near the nominal trajectory. Second, the system tends to be oscillatory, even unstable, due to the "stiffness" of the control law. Thus for a given set of optimization points there exists some lower limit for dT . The tendency toward underdamped response can also be reduced by using an on-line

control update period t_u which is less than the value of dT used in off-line design.

The values of dT used in the performance index for each block is shown in Table 4-1. The fact that dT is not the same throughout state space is explained in the SNM controller discussion.

CM Control Laws

The control law form used in the CM controller design is the hyperellipsoid described in Chapter 2. The control laws in each state space block are thus of the form

$$\begin{aligned} u_1 = & a_1 + a_2x_1 + a_3x_1^2 + a_4x_2 + a_5x_2^2 + a_6x_2x_1 \\ & + a_7x_3 + a_8x_3^2 + a_9x_3x_2 + a_{10}x_3x_1 + a_{11}x_4 \\ & + a_{12}x_4^2 + a_{13}x_4x_3 + a_{14}x_4x_2 + a_{15}x_4x_1 \end{aligned} \quad (4-14)$$

$$\begin{aligned} u_2 = & b_1 + b_2x_1 + b_3x_1^2 + b_4x_2 + b_5x_2^2 + b_6x_2x_1 \\ & + b_7x_3 + b_8x_3^2 + b_9x_3x_2 + b_{10}x_3x_1 + b_{11}x_4 \\ & + b_{12}x_4^2 + b_{13}x_4x_3 + b_{14}x_4x_2 + b_{15}x_4x_1 \end{aligned} \quad (4-15)$$

$$\begin{array}{ll} \text{where } u_1 = S & x_2 = B \\ u_2 = \theta_{bcv} & x_3 = P_{15} \\ x_1 = T_{15} & x_4 = N \end{array}$$

The control law coefficients found from the off-line optimizations for the CM controller are given in Table 4-3.

Table 4-3. CM Control Law Tabulation

BLOCK 1	1.460E+03	1.500E+02	8.400E+01	1.037E+04
1.723E+01	2.102E-02	1.795E-05	-5.042E-02	-1.100E-05
-7.035E-05	-7.516E-02	1.989E-03	1.279E-03	-3.422E-04
2.756E-04	-2.549E-08	1.137E-06	4.043E-06	1.214E-07
2.573E+01	-3.784E-02	4.191E-05	2.037E-03	-1.152E-05
1.915E-05	3.568E-02	-2.760E-04	-9.051E-05	5.136E-05
2.833E-03	1.470E-08	-5.352E-06	-1.556E-06	-3.675E-06
BLOCK 2	1.980E+03	1.500E+02	1.320E+02	1.306E+04
2.638E+01	2.314E-02	8.266E-06	-2.939E-02	-1.560E-06
-1.953E-05	-5.726E-02	7.890E-04	3.898E-04	-1.309E-04
2.554E-04	-1.209E-08	1.830E-06	1.728E-06	-7.575E-08
1.974E+01	-1.692E-02	1.335E-05	1.925E-03	-6.100E-07
2.526E-06	2.363E-02	-6.826E-05	-2.817E-05	3.616E-06
1.528E-03	-9.732E-09	-1.817E-06	-3.185E-07	-1.199E-06
BLOCK 3	2.500E+03	1.500E+02	1.800E+02	1.523E+04
3.682E+01	2.698E-02	7.039E-06	-3.059E-02	-1.733E-06
-1.818E-05	-6.974E-02	6.998E-04	3.151E-04	-1.148E-04
3.057E-04	-1.493E-08	1.417E-06	1.226E-06	6.312E-09
1.682E+01	-1.255E-02	7.822E-06	1.785E-03	-6.764E-07
2.898E-06	2.647E-02	-4.231E-05	-3.901E-05	-3.644E-06
1.574E-03	3.094E-09	-8.486E-07	2.814E-07	-1.027E-06
BLOCK 4	3.020E+03	1.500E+02	2.280E+02	1.709E+04
4.878E+01	2.888E-02	6.074E-06	-2.392E-02	-1.429E-06
-9.685E-06	-6.094E-02	5.346E-04	1.773E-04	-8.786E-05
2.964E-04	1.174E-08	9.216E-07	9.367E-07	-1.415E-07
1.520E+01	-8.297E-03	3.147E-06	1.557E-03	-7.458E-07
1.068E-06	2.105E-02	-9.483E-05	-1.338E-05	9.949E-06
1.163E-03	-2.988E-08	-3.251E-07	3.566E-08	-4.027E-07
BLOCK 5	3.540E+03	1.500E+02	2.760E+02	1.875E+04
6.172E+01	3.342E-02	6.555E-06	-3.232E-02	-1.075E-06
-1.307E-05	-8.921E-02	6.334E-04	2.162E-04	-1.078E-04
4.015E-04	-8.476E-09	1.729E-06	1.102E-06	-7.575E-08
1.418E+01	-8.930E-03	3.372E-06	1.392E-03	2.601E-07
4.077E-07	2.812E-02	-4.254E-05	-8.238E-06	1.688E-06
1.599E-03	9.882E-09	-1.126E-06	-7.413E-08	-6.116E-07

Table 4-3, Continued

BLOCK 6	3.900E+03	1.500E+02	3.333E+02	2.079E+04
7.963E+01	1.025E-01	2.338E-05	-1.724E-02	-1.791E-06
-8.518E-06	-6.818E-02	3.748E-04	1.063E-04	-1.764E-04
-1.139E-04	-9.292E-09	3.314E-06	8.401E-07	-7.495E-07
9.110E+00	-4.694E-02	4.621E-05	-4.332E-03	8.591E-07
6.807E-06	1.714E-02	1.401E-05	8.009E-06	5.123E-06
1.445E-03	1.716E-08	-1.753E-06	-2.864E-07	-2.136E-06
BLOCK 7	4.100E+03	1.500E+02	4.000E+02	2.317E+04
9.571E+01	1.004E-01	-1.170E-08	-1.190E-02	-1.505E-05
-1.174E-05	-5.008E-02	1.833E-04	4.870E-05	-1.963E-05
-1.585E-04	3.183E-08	-3.651E-06	1.237E-06	4.919E-07
4.964E+00	-3.654E-02	3.709E-05	-3.321E-03	6.785E-06
9.449E-06	1.618E-02	2.297E-05	9.479E-06	-3.713E-05
1.108E-03	-8.331E-09	1.744E-06	-4.959E-07	-1.845E-06
BLOCK 8	4.250E+03	0.	4.500E+02	2.485E+04
9.885E+01	-3.931E-02	6.921E-05	-2.511E-02	9.266E-07
-6.800E-06	-5.611E-02	4.762E-04	7.484E-05	-2.338E-04
8.606E-04	2.758E-08	1.698E-06	1.027E-06	3.330E-07
3.521E+00	-2.403E-02	-2.125E-05	-1.979E-03	-5.595E-07
2.406E-06	1.647E-02	-1.054E-04	1.039E-05	1.411E-04
9.230E-04	-1.437E-08	-1.855E-06	-5.960E-07	-8.120E-07
BLOCK 9	4.137E+03	-1.500E+02	4.125E+02	2.360E+04
7.993E+01	1.025E-01	-2.996E-06	-9.119E-03	-1.668E-05
-9.059E-06	-4.809E-02	6.369E-05	6.553E-05	6.918E-06
-2.668E-04	-5.582E-08	4.519E-07	1.929E-06	6.917E-07
6.887E+00	-3.994E-02	4.089E-05	-3.873E-03	8.008E-06
7.893E-06	1.613E-02	6.811E-05	-2.778E-06	-4.383E-05
1.207E-03	3.371E-08	-2.314E-07	-8.021E-07	-2.097E-06
BLOCK 10	3.912E+03	-1.500E+02	3.375E+02	2.094E+04
6.150E+01	1.042E-01	3.105E-06	-1.524E-02	-1.834E-05
1.870E-05	-6.347E-02	2.473E-04	6.348E-05	-9.097E-05
-2.347E-04	-2.718E-08	1.196E-06	-5.451E-07	3.055E-07
1.233E+01	-5.439E-02	7.421E-05	-5.138E-03	1.059E-05
-8.355E-06	1.624E-02	6.390E-05	3.041E-05	-4.201E-05
1.681E-03	3.467E-08	-4.793E-07	4.736E-07	-3.581E-06

Table 4-3, Continued

BLOCK 11	3.540E+03	-1.500E+02	2.760E+02	1.875E+04
4.895E+01	3.491E-02	4.827E-07	-3.164E-02	-1.107E-05
-1.829E-05	-1.020E-01	2.859E-05	3.018E-04	-5.970E-07
2.721E-04	-1.956E-07	-3.311E-06	1.672E-07	1.234E-06
1.357E+01	-9.345E-03	9.713E-06	1.495E-03	7.629E-06
3.862E-06	3.088E-02	5.188E-04	-7.340E-05	-1.069E-04
1.687E-03	1.472E-07	5.749E-06	6.151E-07	-1.911E-06
BLOCK 12	3.020E+03	-1.500E+02	2.280E+02	1.709E+04
3.558E+01	2.995E-02	6.007E-06	-2.301E-02	4.046E-06
-1.068E-05	-7.140E-02	5.856E-04	1.782E-04	-9.977E-05
1.757E-04	7.245E-09	2.076E-06	1.491E-06	-1.206E-07
1.444E+01	-8.580E-03	3.705E-06	1.448E-03	-2.665E-06
1.244E-06	2.341E-02	-8.680E-05	-1.403E-05	1.401E-05
1.250E-03	-1.555E-08	-7.087E-07	-4.818E-07	-6.251E-07
BLOCK 13	2.500E+03	-1.500E+02	1.800E+02	1.523E+04
2.373E+01	2.884E-02	8.328E-06	-2.950E-02	-1.888E-06
-1.760E-05	-8.541E-02	8.358E-04	2.980E-04	-1.375E-04
1.342E-04	5.759E-09	3.261E-06	1.430E-06	-3.277E-07
1.597E+01	-1.349E-02	7.587E-06	2.191E-03	1.061E-06
1.744E-06	3.013E-02	-6.900E-05	-3.108E-05	6.408E-06
1.765E-03	-1.131E-08	-1.654E-06	-2.757E-07	-9.924E-07
BLOCK 14	1.980E+03	-1.500E+02	1.320E+02	1.306E+04
1.300E+01	2.547E-02	9.461E-06	-2.757E-02	6.093E-06
-1.744E-05	-7.583E-02	1.028E-03	3.733E-04	-1.602E-04
5.485E-05	1.516E-08	3.778E-06	1.241E-06	-4.324E-07
1.889E+01	-1.944E-02	1.473E-05	2.626E-03	-1.888E-05
-3.490E-06	3.047E-02	-2.262E-04	-6.817E-05	2.328E-05
1.871E-03	-5.182E-08	-1.837E-06	6.899E-07	-1.372E-06
BLOCK 15	1.460E+03	-7.000E+01	8.400E+01	1.037E+04
8.654E+00	4.734E-03	5.660E-06	-2.517E-02	-4.124E-06
-1.965E-05	-6.245E-02	1.220E-03	4.786E-04	-6.954E-05
1.222E-04	-1.405E-09	2.071E-06	1.721E-06	9.083E-08
2.568E+01	-3.078E-02	3.269E-05	-1.000E-03	1.801E-06
1.026E-05	1.601E-02	-1.111E-04	3.367E-05	3.891E-05
1.531E-03	-6.612E-08	-2.697E-06	-1.554E-06	-1.601E-06

Listed for each block are (1) the state space coordinates $\underline{x}_0$ of the block center, (2) the 15 coefficients by rows describing the S law, and (3) the 15 coefficients by rows describing the θ_{bcv} law.

The vector $\underline{x}_0$ is a part of each law. The control law coefficients are not compatible with absolute state coordinates, but with relative ones:

$$\underline{x}_r = \underline{x} - \underline{x}_0 \quad (4-16)$$

Here, $\underline{x}_r$ is used directly with the 30 coefficients to find the control vector (S, θ_{bcv}) . This coordinate offset is used to improve the conditioning of the fit matrices in the least squares fit process.

Regarding the "goodness" of the hyperellipsoidal fit to the optimization data, the maximum error between the fit and the individual data points is typically two to four orders of magnitude less than the block average, for both laws.

In any given block, some control law terms may be less important than others in describing the optimal law accurately. This is demonstrated in Table 4-4 which gives the percent of the total control contributed by each of the 15 terms for test points in blocks 1 through 8. The test points are selected to maximize each relative state within the block. The table shows that certain terms have almost negligible effect in all eight blocks. For instance terms 5, 12, 13, and 15 in the S law, and terms 5, 9, and 12 in

Table 4-4. Contribution of CM Control Law Terms in Percent of Total; S Law Top,
 θ_{bcv} Law Bottom

TERM	BLOCK 1	BLOCK 2	BLOCK 3	BLOCK 4	BLOCK 5	BLOCK 6	BLOCK 7	BLOCK 8
1	81.672	86.776	89.817	90.677	92.827	92.598	93.313	101.640
2	25.899	19.795	17.112	13.959	13.070	11.916	9.794	2.021
3	5.750	1.838	1.161	.763	.667	.272	.000	.178
4	11.948	4.835	3.731	2.223	2.431	1.002	.580	.645
5	.130	.013	.011	.007	.004	.005	.037	.001
6	4.334	.835	.576	.234	.256	.050	.057	.009
7	15.673	8.289	7.486	4.984	5.904	4.228	2.604	2.115
8	18.247	5.025	3.305	1.924	1.844	1.240	.508	.658
9	13.335	2.821	1.691	.725	.715	.330	.127	.071
10	18.554	4.926	3.205	1.868	1.855	1.094	.102	.441
11	3.179	1.791	1.463	1.020	1.069	.293	.329	1.364
12	.716	.181	.140	.075	.040	.053	.141	.067
13	.577	.565	.298	.140	.203	.455	.404	.099
14	2.332	.606	.293	.161	.147	.108	.128	.041
15	.364	.138	.008	.127	.052	.193	.102	.026

TERM	BLOCK 1	BLOCK 2	BLOCK 3	BLOCK 4	BLOCK 5	BLOCK 6	BLOCK 7	BLOCK 8
1	105.611	100.702	95.049	94.965	89.715	110.139	112.617	82.492
2	40.389	22.441	18.448	13.474	14.693	56.746	82.897	28.157
3	11.629	4.604	2.988	1.329	1.443	5.587	8.415	1.245
4	.418	.491	.504	.486	.440	2.619	3.767	1.159
5	.118	.008	.010	.012	.004	.026	.385	.008
6	1.022	.168	.213	.087	.034	.411	1.072	.070
7	6.444	5.306	6.582	5.785	7.830	11.051	19.575	14.151
8	2.193	.674	.463	1.147	.521	.482	1.482	3.319
9	.817	.316	.485	.184	.115	.258	.574	.223
10	2.412	.211	.236	.711	.122	.330	4.493	6.060
11	28.300	16.617	17.457	13.449	17.924	38.665	53.512	33.347
12	.357	.226	.067	.639	.196	1.016	.857	.800
13	2.353	.870	.414	.165	.555	2.502	4.493	2.457
14	.777	.173	.156	.021	.042	.383	1.198	.538
15	9.545	3.390	2.962	1.211	1.782	5.715	8.913	1.467

the θ_{bcv} law all contribute less than one percent of the total in each of the eight blocks. Conceivably, the optimization data could be refitted to a control law form which excludes these less important terms without degrading the laws significantly. The on-line control program loses its basic simplicity, however, if the same control law forms are not used for both controls and in all blocks.

Control Model Controller On-Line Operation

The control laws and coordinates contained in Table 4-3 constitute the heart of the CM controller. On-line operation is depicted by the time sequence given in the block diagram of Fig. 4-4. This sequence is repeated every 0.1 seconds.

The control sequence contains a test of control constraint violation which must be explained. In the off-line optimizations leading to the CM control laws, the ribbon widths and performance indices were chosen to avoid the physical control constraints,

$$0 \leq S \leq 120$$

$$0 \leq \theta_{bcv} \leq 100$$

On-line, of course, these constraints must be observed. As is shown in Chapter 5, however, nominal operation or even recovery from states quite far from the nominal trajectory do not require controls which violate these constraints.

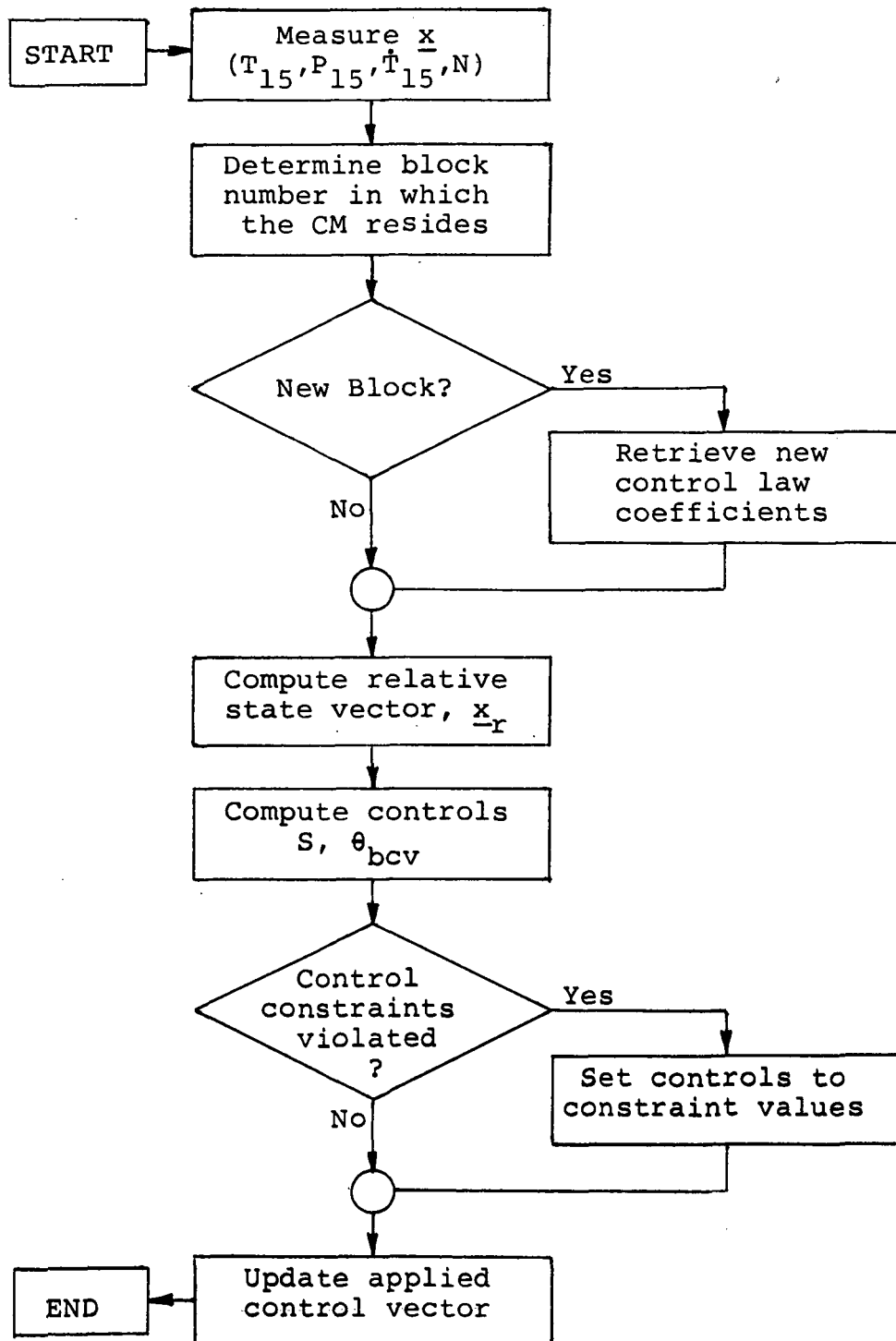


Fig. 4-4. Block Diagram of CM Control Update Sequence

Simplified Nonlinear Model Controller Off-Line Design

Because the off-line design procedure of the SNM and CM controllers are basically the same, only the differences between the two are discussed here. First, it is necessary to modify the trajectory specification used in off-line optimizations. This is due to the neutronics approximations made in the CM development. Second, the dT term in the performance index used in off-line computations has a strong influence on the resulting control laws. Following discussion of these topics, the SNM control laws are presented in tabular form.

Nominal Trajectory Specification

The approximation, $\dot{S} \approx 0$, used in the development of the Control Model, causes the SNM and the CM to react differently during the short transient period following an abrupt change in S or, for the SNM, in DKD . The result is a predictable difference between the temperature rate achieved by the SNM control system and that of the original trajectory. For example, Fig. 4-5 shows the CM controller operating the SNM in block 4. The original trajectory specification calls for a constant B of $150^{\circ}\text{R}/\text{sec}$ in this temperature region, but the actual B varies from about $177^{\circ}\text{R}/\text{sec}$ at 2760°R to $180^{\circ}\text{R}/\text{sec}$ at 3280°R . By modifying the original trajectory specification a proportional amount, the actual desired trajectory is obtained as shown in

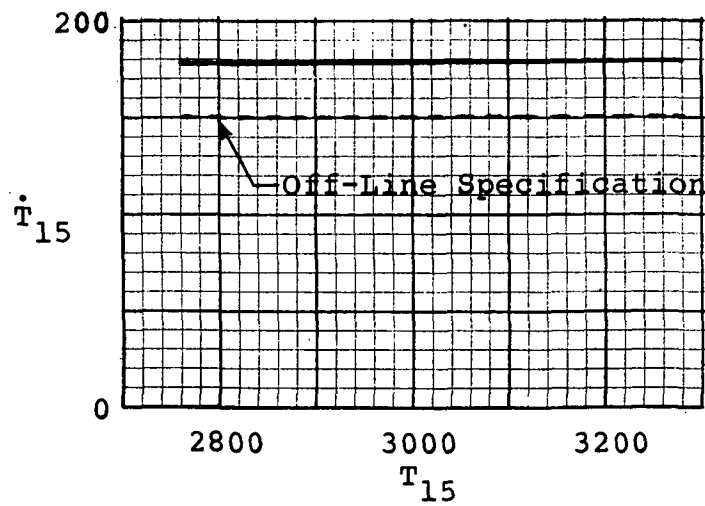


Fig. 4-5. SNM Control System Response to CM Controller, Block 4

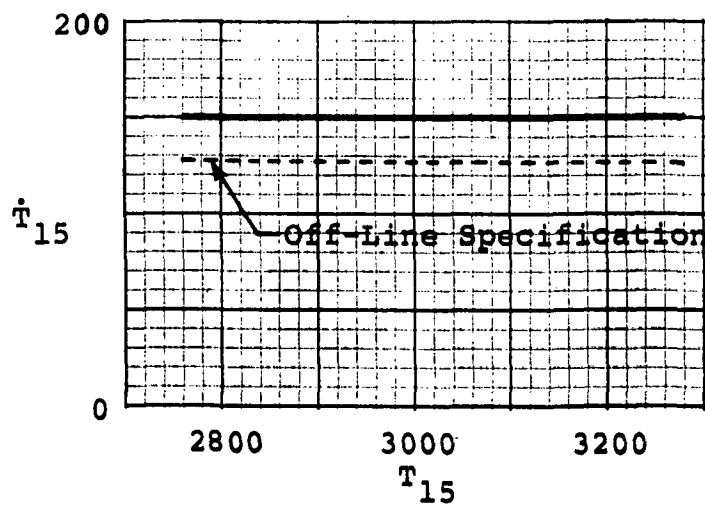


Fig. 4-6. SNM Control System Response to SNM Controller, Block 4

Fig. 4-6. The amount of offset depends upon the value of the nuclear power level, the hydrogen flow rate, etc., and is different for each block.

It is interesting to note that the (T_{15}, P_{15}) portion of the SNM trajectory is unaffected by any of the approximations made in the development of the CM; it is not necessary to modify the trajectory specification in this plane.

Performance Index

The same performance indices are used in the off-line optimizations which define the CM and SNM controllers. Table 4-1 indicates that the value of dT , part of the performance index, is not constant throughout applicable state space. The reason for this is as follows. Near Idle Point, the nuclear power level is very low and the various thermodynamic time constants in the engine system are relatively large. Thus the system is more sluggish and requires more time to recover from a given state error. To accommodate this characteristic, dT is made larger in the blocks which include or are near Idle Point and smaller in blocks corresponding to higher values of nuclear power level. However, values of dT less than 0.3 seconds tend to produce control laws which magnify the differences in the short-term transient behavior of the CM and the SNM.

SNM Control Laws

With the exception of the trajectory modification discussed above, the SNM control laws are obtained in exactly the same manner as were the CM control laws. These laws are given in Table 4-5 with entries defined the same as in Table 4-3.

Simplified Nonlinear Model Controller On-Line Operation

The on-line operation of the SNM controller is essentially the same as that of the CM controller with three important differences. First, B is not a state in the SNM and hence must be approximated on-line. Second, the basic controller supplies the pseudo control S which must be converted to the actual control DKD . Third, the input reactivity DKD is subject to physical magnitude and rate constraints. These topics are discussed in this section.

On-Line Estimation of B

In the CM, B exists as a state and, since the SNM controller design is based on the CM, the SNM controller must be supplied on-line measurements of B . Since B does not exist explicitly in the SNM it must be estimated by a sequence of T_{15} measurements. In this application T_{15} is assumed to be piecewise quadratic in t and the value of $B(t)$ approximated as a function of $T_{15}(t)$, $T_{15}(t - t_u/2)$, and $T_{15}(t - t_u)$ where t_u is the nominal control update period.

Table 4-5. SNM Control Law Tabulation

BLOCK 1	1.460E+03	1.278E+02	8.400E+01	1.037E+04
1.631E+01	2.081E-02	1.546E-05	-5.002E-02	-2.053E-05
-6.947E-05	-7.932E-02	1.973E-03	1.253E-03	-3.277E-04
2.544E-04	-5.952E-08	1.512E-06	3.947E-06	4.685E-07
2.555E+01	-3.854E-02	5.206E-05	2.128E-03	2.758E-05
1.638E-05	3.603E-02	1.000E-04	-5.769E-05	-4.448E-06
2.902E-03	1.518E-07	-5.743E-06	-1.565E-06	-5.153E-06
BLOCK 2	1.980E+03	1.244E+02	1.320E+02	1.306E+04
2.524E+01	2.287E-02	7.507E-06	-2.924E-02	-2.111E-06
-2.071E-05	-5.936E-02	7.894E-04	3.941E-04	-1.275E-04
2.442E-04	-2.350E-08	2.122E-06	1.885E-06	5.662E-08
1.966E+01	-1.715E-02	1.496E-05	1.711E-03	-1.084E-07
5.312E-06	2.454E-02	-2.856E-05	-4.205E-05	-3.805E-06
1.547E-03	1.414E-08	-2.100E-06	-6.274E-07	-1.505E-06
BLOCK 3	2.500E+03	1.279E+02	1.800E+02	1.523E+04
3.587E+01	2.682E-02	6.177E-06	-3.058E-02	-1.518E-05
-1.657E-05	-7.083E-02	6.745E-04	2.947E-04	-1.125E-04
2.988E-04	-4.310E-08	2.413E-06	1.476E-06	1.286E-07
1.673E+01	-1.262E-02	9.003E-06	1.805E-03	1.874E-05
5.399E-07	2.653E-02	1.093E-05	-4.555E-06	-7.327E-06
1.584E-03	4.183E-08	-2.208E-06	-2.986E-07	-1.170E-06
BLOCK 4	3.020E+03	1.260E+02	2.280E+02	1.709E+04
4.773E+01	2.866E-02	5.241E-06	-2.381E-02	3.193E-06
-1.137E-05	-6.163E-02	4.379E-04	2.014E-04	-7.165E-05
2.923E-04	-9.884E-09	2.009E-06	1.534E-06	-1.464E-07
1.514E+01	-8.271E-03	3.982E-06	1.489E-03	-4.208E-06
2.748E-06	2.108E-02	4.901E-06	-3.018E-05	-5.798E-06
1.167E-03	-4.376E-09	-1.261E-06	-6.047E-07	-4.363E-07
BLOCK 5	3.540E+03	1.319E+02	2.760E+02	1.875E+04
6.096E+01	3.354E-02	5.789E-06	-3.230E-02	-1.229E-05
-1.458E-05	-8.978E-02	6.125E-04	2.115E-04	-1.006E-04
4.009E-04	-1.862E-08	8.009E-07	1.255E-06	1.113E-07
1.414E+01	-8.921E-03	3.946E-06	1.399E-03	7.481E-06
9.883E-07	2.806E-02	-2.467E-05	-2.969E-06	-4.229E-06
1.600E-03	1.774E-08	-5.604E-07	-1.588E-07	-7.355E-07

Table 4-5, Continued

BLOCK 6	3.900E+03	7.500E+01	3.333E+02	2.079E+04
7.482E+01	9.895E-02	2.616E-05	-1.742E-02	-1.020E-06
-1.293E-05	-6.838E-02	3.651E-04	1.044E-04	-1.726E-04
-1.172E-04	-7.061E-09	3.423E-06	1.081E-06	-9.618E-07
9.949E+00	-4.855E-02	4.684E-05	-4.468E-03	3.372E-07
9.299E-06	1.725E-02	1.589E-05	1.015E-05	7.988E-06
1.506E-03	1.663E-08	-1.879E-06	-4.755E-07	-2.142E-06
BLOCK 7	4.100E+03	6.500E+01	4.000E+02	2.317E+04
9.032E+01	9.708E-02	1.937E-05	-1.166E-02	-1.415E-07
-2.564E-06	-5.069E-02	2.206E-04	5.681E-05	-1.036E-04
-1.329E-04	-7.096E-09	2.542E-06	6.543E-07	-7.353E-07
5.706E+00	-3.756E-02	2.773E-05	-3.343E-03	1.598E-07
3.801E-06	1.624E-02	7.040E-06	9.693E-06	6.576E-06
1.135E-03	7.875E-09	-1.303E-06	-2.345E-07	-1.223E-06
BLOCK 8	4.250E+03	0.	4.500E+02	2.485E+04
9.885E+01	2.840E-02	4.250E-05	-1.692E-02	4.566E-06
1.178E-05	-4.904E-02	2.436E-04	5.366E-05	-8.482E-05
3.710E-04	6.490E-09	1.226E-06	3.830E-07	-7.776E-07
3.519E+00	-2.800E-02	1.004E-06	-2.317E-03	-2.199E-06
-4.970E-06	1.612E-02	-1.392E-05	1.025E-05	3.335E-05
9.323E-04	-1.986E-09	-1.083E-06	-1.549E-07	-4.470E-07
BLOCK 9	4.137E+03	-8.640E+01	4.125E+02	2.360E+04
8.394E+01	1.000E-01	1.725E-05	-1.018E-02	1.333E-06
-5.685E-06	-4.770E-02	1.629E-04	1.244E-04	-4.676E-05
-2.052E-04	2.858E-08	-7.528E-07	-2.512E-07	-6.882E-07
6.333E+00	-3.867E-02	2.834E-05	-3.439E-03	-1.504E-06
5.715E-06	1.585E-02	1.709E-05	-2.737E-05	-1.439E-05
1.159E-03	-1.249E-08	2.971E-07	1.960E-07	-1.206E-06
BLOCK 10	3.912E+03	-8.190E+01	3.375E+02	2.094E+04
6.581E+01	1.050E-01	3.596E-05	-1.548E-02	8.046E-06
7.357E-06	-6.388E-02	4.181E-04	1.201E-04	-2.113E-04
-2.088E-04	2.328E-08	8.528E-07	-9.692E-07	-9.211E-07
1.151E+01	-5.231E-02	4.769E-05	-4.844E-03	-6.017E-06
-2.710E-06	1.637E-02	-3.645E-05	-5.290E-06	3.667E-05
1.611E-03	2.006E-09	-5.420E-07	7.901E-07	-2.552E-06

Table 4-5, Continued

BLOCK 11	3.540E+03	-1.254E+02	2.760E+02	1.875E+04
4.994E+01	3.621E-02	1.035E-05	-3.167E-02	3.610E-06
-1.295E-05	-1.016E-01	8.889E-04	1.717E-04	-1.874E-04
2.400E-04	-1.751E-07	4.009E-06	2.372E-06	7.373E-07
1.364E+01	-9.494E-03	-2.091E-06	1.607E-03	-5.801E-06
-6.212E-08	3.142E-02	-3.982E-04	1.199E-05	9.601E-05
1.710E-03	3.557E-08	-3.922E-06	-7.235E-07	-5.981E-07
BLOCK 12	3.020E+03	-1.242E+02	2.280E+02	1.709E+04
3.674E+01	3.029E-02	4.517E-06	-2.336E-02	-1.669E-05
-1.401E-05	-7.064E-02	5.088E-04	1.931E-04	-7.954E-05
1.805E-04	-4.194E-08	2.026E-06	1.555E-06	8.170E-08
1.447E+01	-8.554E-03	4.844E-06	1.732E-03	1.335E-05
4.441E-06	2.331E-02	-2.516E-05	-2.952E-05	-4.038E-06
1.244E-03	2.314E-08	-7.011E-07	-5.546E-07	-7.439E-07
BLOCK 13	2.500E+03	-1.274E+02	1.800E+02	1.523E+04
2.473E+01	2.918E-02	7.911E-06	-2.939E-02	-1.105E-05
-1.592E-05	-8.410E-02	9.003E-04	2.805E-04	-1.501E-04
1.369E-04	-1.232E-08	3.252E-06	1.516E-06	-9.617E-08
1.601E+01	-1.345E-02	8.229E-06	2.038E-03	1.414E-05
6.797E-09	2.962E-02	-1.435E-04	-1.297E-05	2.010E-05
1.759E-03	1.122E-08	-2.080E-06	-1.975E-07	-1.237E-06
BLOCK 14	1.980E+03	-1.262E+02	1.320E+02	1.306E+04
1.407E+01	2.598E-02	8.889E-06	-2.743E-02	-3.760E-06
-2.032E-05	-7.385E-02	9.690E-04	3.487E-04	-1.593E-04
5.967E-05	-3.576E-09	4.525E-06	1.815E-06	-3.506E-07
1.891E+01	-1.920E-02	1.581E-05	2.192E-03	6.362E-07
3.772E-06	2.903E-02	-1.499E-04	-2.012E-05	2.098E-05
1.854E-03	-1.022E-08	-3.583E-06	-5.944E-07	-1.535E-06
BLOCK 15	1.460E+03	-7.000E+01	8.400E+01	1.037E+04
8.654E+00	4.734E-03	5.660E-06	-2.517E-02	-4.124E-06
-1.965E-05	-6.245E-02	1.220E-03	4.786E-04	-6.954E-05
1.222E-04	-1.405E-09	2.071E-06	1.721E-06	9.083E-08
2.568E+01	-3.078E-02	3.269E-05	-1.000E-03	1.801E-06
1.026E-05	1.601E-02	-1.111E-04	3.367E-05	3.891E-05
1.531E-03	-6.612E-08	-2.697E-06	-1.554E-06	-1.601E-06

The result is

$$B(t) \approx \frac{T_{15}(t - t_u) - 4T_{15}(t - t_u/2) + 3T_{15}(t)}{t_u} \quad (4-17)$$

Notice that T_{15} must be measured every $t_u/2$ seconds while all other state measurements are necessary only every t_u seconds.

Computation of DKD

On-line measurements other than T_{15} , P_{15} , and N must be made in order to convert the pseudo input S to the actual input DKD. This conversion and the necessary additional measurements, S_a and DKT, are shown in Fig. 4-7. In the figure, block A represents the basic CM-derived controller. This block computes the demand value of θ_{bcv} directly and the demand value of nuclear power level, S_d . The actual value, S_a , is given by SNM equation (3-23) repeated here.

$$S_a = \frac{.0276C_1 + .235C_2 + 1.65C_3}{660(1 - DKT)} \quad (4-18)$$

Now the demand value is obtained by changing DKD and hence DKT by an amount ΔDKD .

$$S_d = \frac{.0276C_1 + .235C_2 + 1.65C_3}{660(1 - DKT - \Delta DKD)} \quad (4-19)$$

By combining equations (4-18) and (4-19) the following is

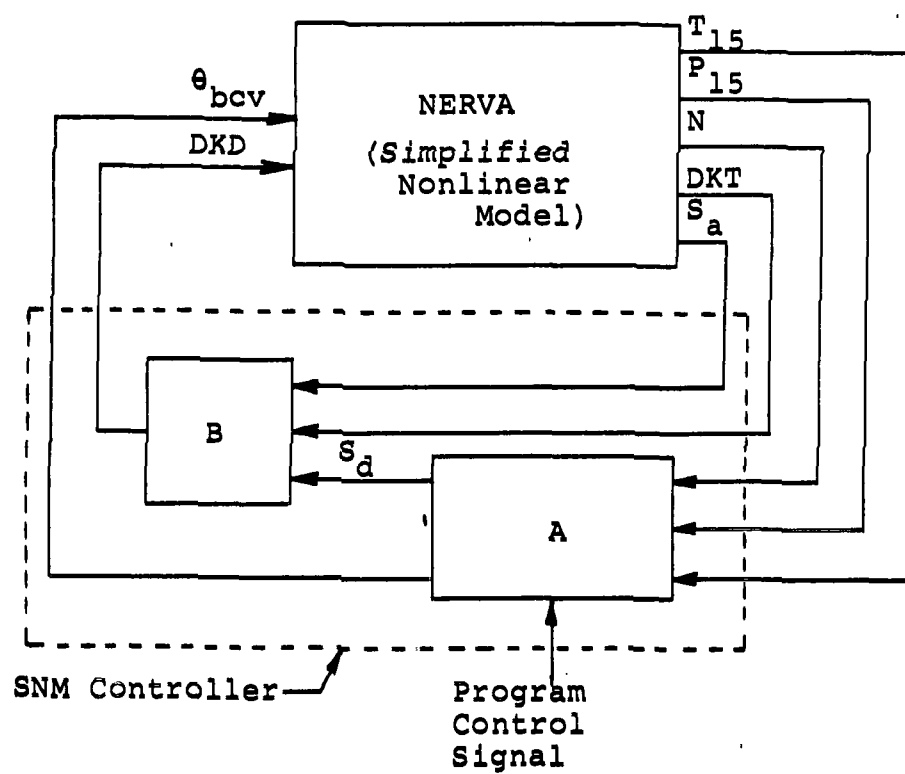


Fig. 4-7. On-Line SNM Control Showing Conversion of Pseudo-Control S_d

obtained.

$$\Delta DKD = (1 - DKT)(1 - S_a/S_d) \quad (4-20)$$

The control DKD is then updated by taking

$$DKD = DKD_{\text{previous}} + \Delta DKD \quad (4-21)$$

It is equations (4-20) and (4-21) which are implemented by block B in the figure. The Program Control Signal simply enables either the startup or shutdown portions of the control law.

DKD Magnitude and Rate Constraints

The SNM as presented by Kendrick does not include certain constraints which are important characteristics of an actual engine. The actual input reactivity for the proposed NERVA engine is of the form

$$DKD = 4.5 \sin^2\left(\frac{\theta_d}{2}\right) - 4.5 \quad (4-22)$$

where θ_d is the input reactivity drum angle, $0 \leq \theta_d \leq 180^\circ$. Now the time rate of change of θ_d is physically constrained to $12^\circ/\text{sec}$ or less, either increasing or decreasing. Thus there exists a constraint on the magnitude of $d(DKD)/dt$ which is a function of θ_d or equivalently of DKD. Taking the time derivative of equation (4-22),

$$\frac{d(DKD)}{dt} = 2.25 \sin(\theta_d) \frac{d\theta_d}{dt} \quad (4-23)$$

$$\text{or,} \quad \left[\frac{d(DKD)}{dt} \right]_{\max} = 2.25 \sin(\theta_d) \left[\frac{d\theta_d}{dt} \right]_{\max} \quad (4-24)$$

Now solve equation (4-22) for θ_d

$$\theta_d = 2 \sin^{-1} \sqrt{\frac{DKD + 4.5}{4.5}} \quad (4-25)$$

and substitute into equation (4-22), noting that the maximum value of $d\theta_d/dt$ is 0.2094 rad/sec.

$$\left[\frac{d(DKD)}{dt} \right]_{\max} = 0.471 \sin \left[2 \sin^{-1} \sqrt{\frac{DKD + 4.5}{4.5}} \right] \quad (4-26)$$

It is not necessary that this rate constraint be actually implemented by the controller since the drum actuators are self-limiting; but it is important to include both the rate and magnitude constraints in any simulations which are conducted. In this study, equations (4-22) and (4-26) are implemented in all SNM digital simulations. The effects of these constraints are noted in Chapter 5.

Summary

Controllers for both the Control Model and the Simplified Nonlinear Model are designed in this chapter using the PSDC method, and the on-line operation of these controllers is explained. Both controllers have the same design basis except that the nominal trajectory specification used

in the SNM controller design is modified slightly. The actual performance of the CM control system and that of the SNM control system is documented in Chapter 5.

CHAPTER 5

CONTROL MODEL AND SIMPLIFIED NONLINEAR MODEL CONTROL SYSTEM PERFORMANCE

Controllers for both the CM and SNM are designed in Chapter 4. In this chapter the performance of the control systems incorporating these controllers is presented. Although the main objective in this study is to demonstrate acceptable SNM control system behavior utilizing the PSDC method, parallel experiments are conducted with the CM control system. Three aspects of both the CM and SNM control system performance are considered. The first is the ability of the system to follow the nominal trajectory. The second is the ability of the controller to return the system to the nominal trajectory; i.e., to recover from initial states which are off the nominal. Third, an attempt is made to determine how sensitive the controller is to changes in plant (model) parameters.

All system simulations are performed digitally using the CDC 6400 Computer. A fourth order Runge-Kutta integration routine is used with a 0.05 second integration step size. The control update period t_u is 0.1 seconds in all cases. A typical FORTRAN simulation program for the SNM system is given in Appendix B.

Control Model Control System Performance

A brief analysis of the CM control system performance is certainly justified. Such an analysis serves as a reference by which to compare the SNM system performance. Also, this particular PSDC application is of interest because it represents a case where an exact model of the plant to be controlled is available. Hence it is possible to examine the advantages and failings of the method itself without clouding the issue with modeling inaccuracies.

Nominal CM Control System Performance

The ability of the CM controller to maintain the CM on the nominal specified trajectory is examined here. The specification used in designing the controller is the start-up and shutdown trajectories developed in Chapter 4. Fig. 5-1 shows time plots of the states T_{15} , P_{15} , and B , for startup and for shutdown. Deviations from an ideal profile are noticeable only in the B plot. The exponential behavior near Idle and Design Points are normal and agree with the original specification. Between these points, however, B should be $+150^{\circ}\text{R/sec}$ during startup and -150°R/sec during shutdown. Error as high as 13 percent is evident, especially where P_{15} is changing rapidly. This discrepancy is considered minor and is due to the fact that $dT \neq t_u$. Experiments show that if $dT = 0.1$ in the off-line design, then $dT = t_u$ and the CM follows the nominal trajectory precisely.

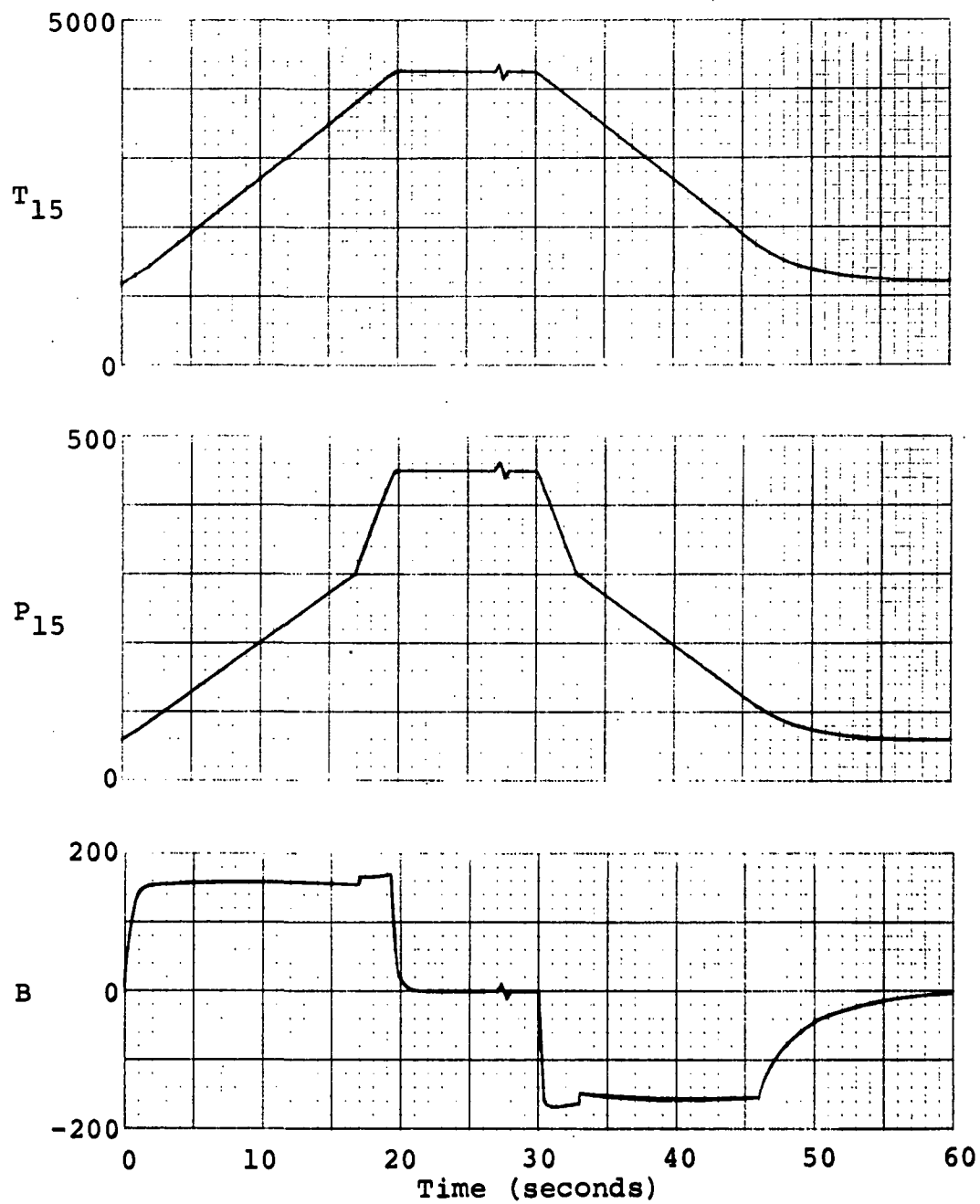


Fig. 5-1. CM Control System Startup/Shutdown Time Response

These small errors could also be eliminated by retaining the various values of dT as presently used but off-setting the B trajectory specification slightly as was done in the SNM design.

CM Control System Recovery

An important characteristic of the CM controller is its ability to recover, or to return the plant to the nominal trajectory from various points in state space. Fig. 5-2 shows trajectories in the (T_{15}, B) plane corresponding to ± 100 and -100 percent initial error in B . Both converge smoothly to the nominal. Apparently recovery is possible from any initial value of B . Of course for $B(0) \gg 300^\circ R/\text{sec}$ or $B(0) \ll 0^\circ R/\text{sec}$, the constraints on the pseudo-input S will increase the recovery time.

Fig. 5-3 shows trajectories in the (T_{15}, P_{15}) plane which correspond to initial errors of $(+50 \text{ psi}, +2500 \text{ rpm})$ and $(-50 \text{ psi}, -2500 \text{ rpm})$ in P_{15} and N respectively. Again, recovery is smooth and there is no sign of instability. An attempt was made to determine a region in the (T_{15}, P_{15}) plane wherein recovery to the nominal is possible. Experiments indicate however, that the CM control system is capable of recovery from any point in that plane which is physically attainable in view of the constraints on S and θ_{bcv} .

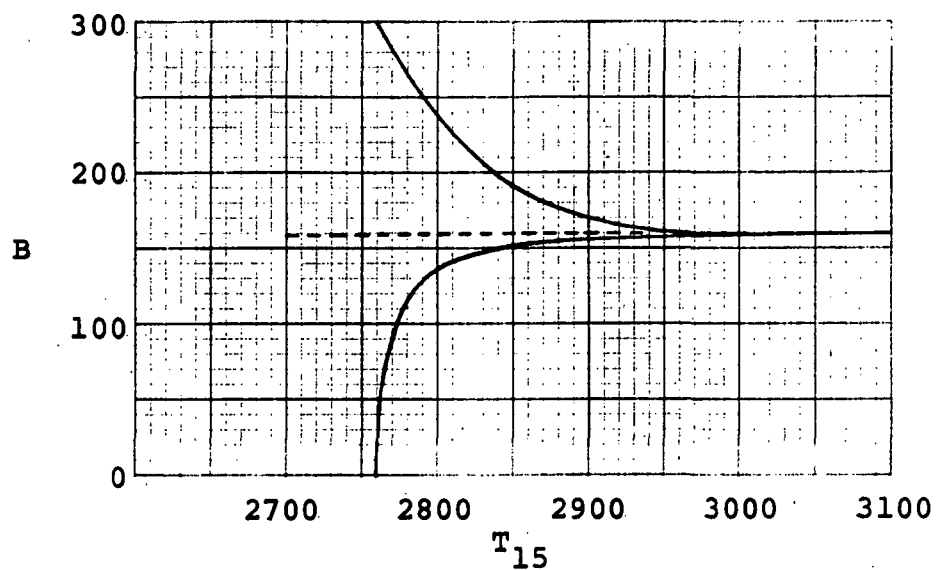


Fig. 5-2. CM Control System Recovery Trajectories in the (T_{15}, B) Plane

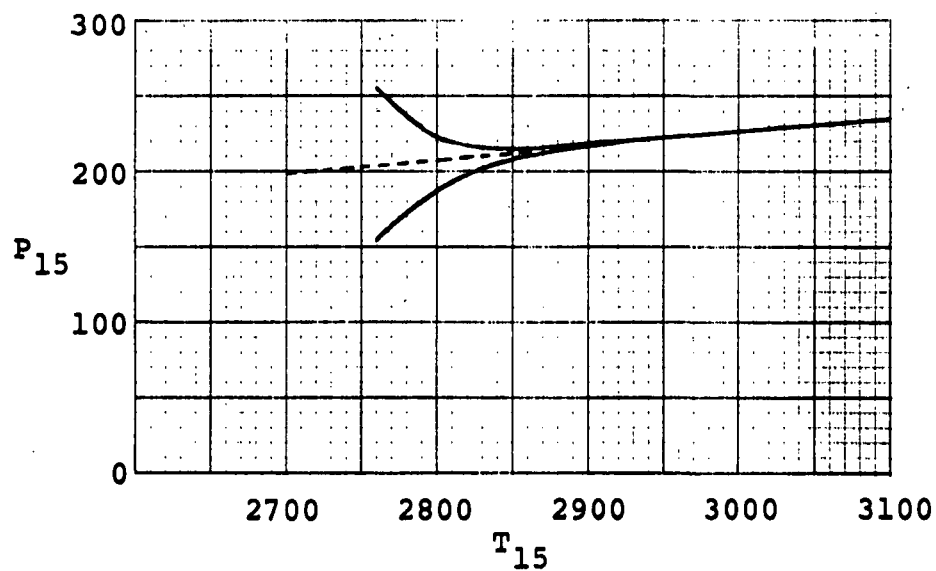


Fig. 5-3. CM Control System Recovery Trajectories in the (T_{15}, P_{15}) Plane

CM Controller Sensitivity

In a conventional control system, time-varying reference inputs are used to control the plant in some fashion. With the PSDC method, however, the desired system behavior is embedded in the controller itself. No external signals are required except possibly to utilize different controllers corresponding to different desired trajectories. Because a PSDC system operates so independently, it is important to consider the effects of variations in plant parameters arising either from aging processes or simply from model inaccuracies.

To test the sensitivity of the CM controller to changes in plant parameters, startup simulations are conducted in which a dominant CM time constant is changed. Fig. 5-4 shows startup state space trajectory errors relative to the ideal specification for the CM control system in which each coefficient in the turbopump equation (3-28) is changed by +25 and -25 percent. The unperturbed model startup trajectory errors are also shown for comparison. State space error plots are given instead of full scale time plots, because these small errors would be obscured in the latter.

The system is seen to be quite insensitive to these perturbations over most of the T_{15} excursion. From Idle to Throttle Point ($T_{15} = 3800$), temperature rate error is less than $3^{\circ}\text{R}/\text{sec}$ and pressure error is less than 1 psi. The

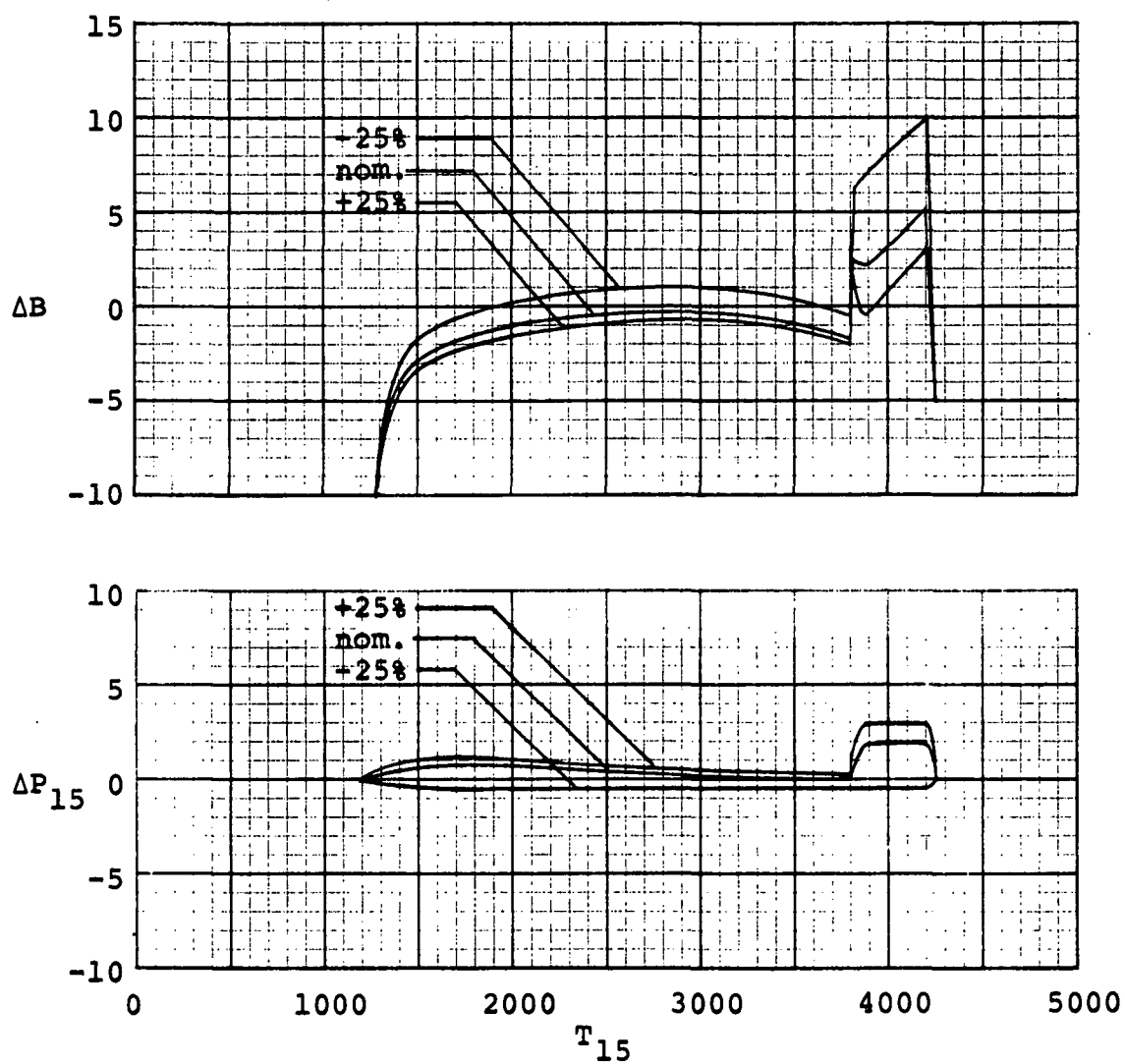


Fig. 5-4. CM Control System Startup Trajectory Errors due to Changes in Turbopump Equation Coefficients

effects of the perturbations are more noticeable during the short interval from Throttle to Design Point but are still considered minor.

Simplified Nonlinear Model Control System Performance

The primary goal of this study is to demonstrate that the SNM can be controlled in an acceptable manner using the PSDC method, even though the controller design is based on the lower order CM. This is the case, as is verified by the data presented here.

Nominal SNM Control System Performance

The most important characteristic of the SNM controller is its ability to maintain the SNM on the nominal trajectory. The nominal specification is the same startup/shutdown trajectory used in the design of the CM controller, except for the slight modifications in the (T_{15} , B) plane as described in Chapter 4.

Startup and shutdown time plots for states T_{15} , P_{15} , and B are shown in Fig. 5-5. The responses are smooth and deviate from the ideal very little. At startup, B increases to 150°R/sec more slowly than did the CM system. This is due to the DKD rate constraint explained earlier. The temperature rate B follows the nominal trajectory within one percent over most of the startup/shutdown cycle. The error rises to a maximum of 15 percent just before entering the

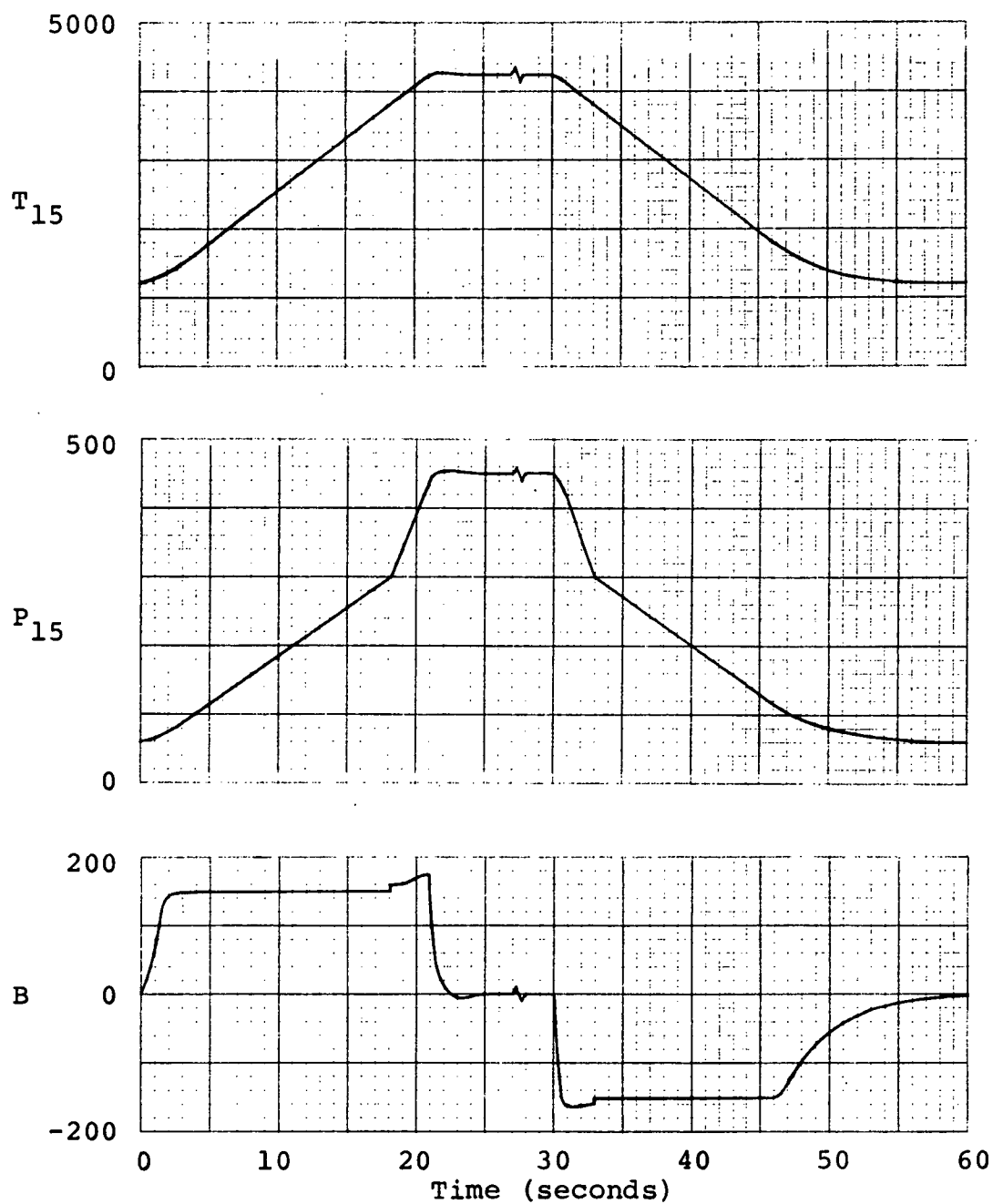


Fig. 5-5. SNM Control System Startup/Shutdown Time Response

regulation block at Design Point. Though not easily seen on the plots, T_{15} and P_{15} exhibit a slight overshoot of 14°R and 3 psi, both well below one percent, as Design Point is approached.

As the system approaches Design Point, B decays to zero rapidly, beginning 50°R below Design temperature. But when approaching Idle Point, B decays more slowly, beginning 520°R above Idle temperature. This behavior is normal and corresponds to the nominal trajectory specification of Fig. 4-3. The reason for this unsymmetrical specification should be explained.

A system startup at the rate of $150^{\circ}\text{R}/\text{sec}$ to within 50°R of Design Point requires a DKD excursion of about one dollar. However, for a system shutdown at $-150^{\circ}\text{R}/\text{sec}$ to within 50°R of Idle Point, a DKD excursion of about -11 dollars is required. This large amount of poisoning is required to cancel the precursor reactivity which decays very slowly from the high level corresponding to Design Point. Now equation (4-22) indicates a lower constraint of -4.5 dollars for DKD and hence shutdown at $-150^{\circ}\text{R}/\text{sec}$ cannot be sustained below about 1700°R . This constraint is avoided by specifying a shutdown trajectory which holds B at $-150^{\circ}\text{R}/\text{sec}$ to 1720°R and then forces B to decay slowly toward zero from there.

SNM Control System Recovery

Data are presented here which describe the ability of the SNM control system to return to the nominal trajectory when started at points in state space off the nominal.

Because of the structure and complexity of the SNM, it is not feasible to initialize the system in a dynamic condition. In fact, it is necessary to solve a set of simultaneous nonlinear equations even to obtain static model data. Therefore, starting the system anywhere in the (T_{15}, P_{15}) plane other than at Idle or Design Points (where $B = 0$) necessarily requires a "recovery" in the state B .

Fig. 5-6 shows trajectories in the (T_{15}, P_{15}) and (T_{15}, B) planes which correspond to initial P_{15} errors of +50 psi and -50 psi. For the +50 psi case, the system recovers via a strange looking trajectory; B actually goes negative for a short period while the pressure drops rapidly. This is caused by the abrupt loss of reactivity from hydrogen flow, and the fact that the time rate of change of the input reactivity, DKD , is constrained.

Numerous experiments were conducted to determine an approximate initial condition region in the (T_{15}, P_{15}) plane wherein recovery to the nominal trajectory is possible. Because of the stringent magnitude and rate constraints on DKD this region is not as large as it is for the CM control system. It can be roughly described as a strip at least 150 psi wide and centered on the nominal trajectory. In other

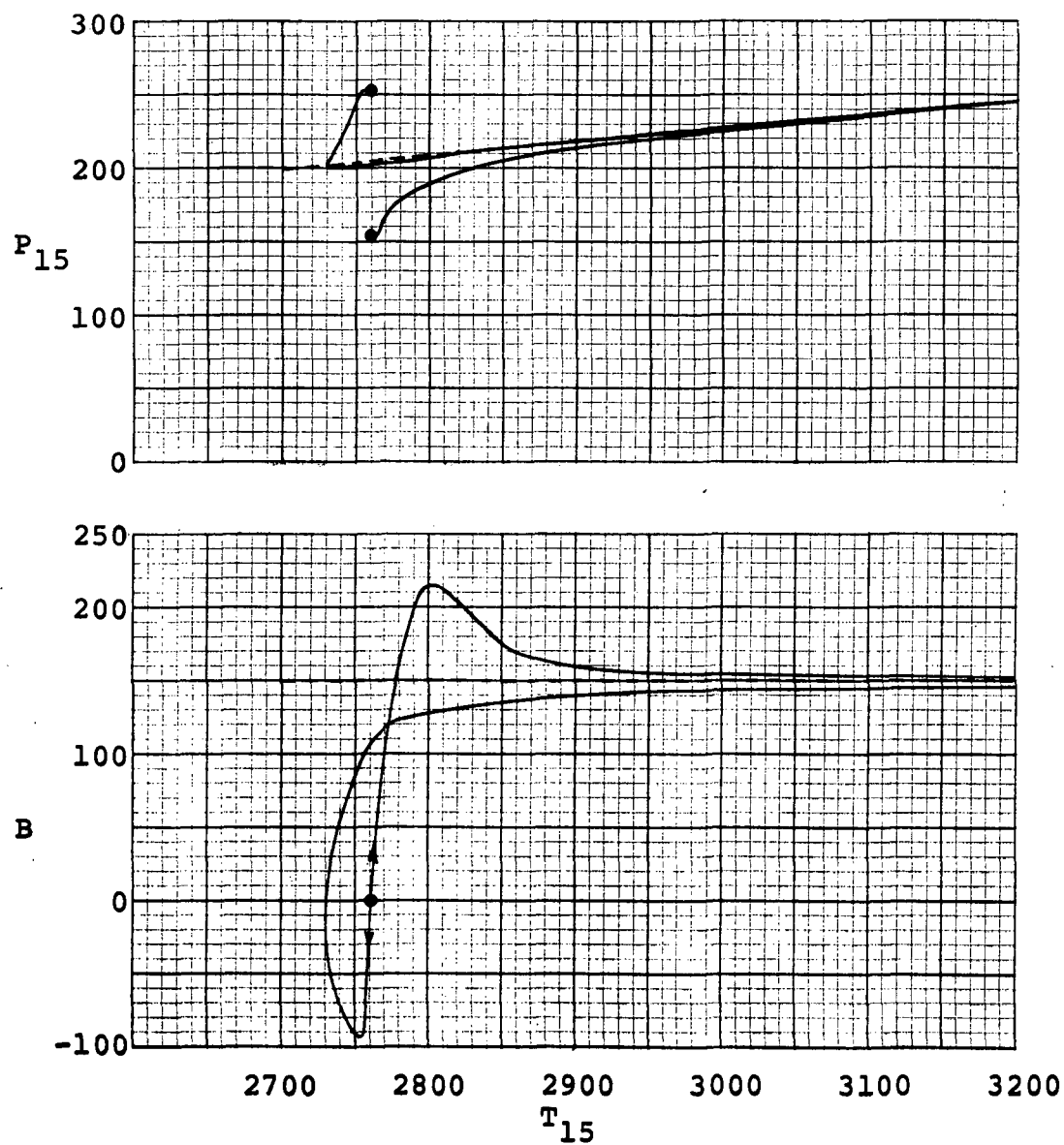


Fig. 5-6. SNM Control System Recovery Trajectories in the (T_{15}, P_{15}) and (T_{15}, B) Planes

words initial pressure errors between +75 and -75 psi can be tolerated anywhere along the temperature axis providing such errors correspond to points in the (T_{15}, P_{15}) plane which are physically realizable.

SNM Controller Sensitivity

The sensitivity of the CM controller to changes in its plant (the Control Model) is briefly examined in a previous section. The sensitivity of the SNM controller to changes in its plant (the Simplified Nonlinear Model) is of more vital concern because of the analytical "gap" which already exists between the SNM and its controller. SNM controller sensitivity is tested by varying both the turbopump assembly and heat transfer time constants in the SNM and examining the resulting system startup responses.

Fig. 5-7 shows P_{15} error relative to the ideal startup trajectory and the B startup trajectory for the SNM control system in which the coefficients in the turbopump equation (3-20) are changed by +25 and -25 percent. The unperturbed model trajectory is omitted for clarity, and would fall between the two curves in either plot. Deviation from the normal startup trajectory due to these turbopump parameter variation is seen to be very minor.

The effects of +25 and -25 percent variations in all coefficients of the SNM temperature equations (3-1), (3-2), (3-3), (3-21), and (3-22) are shown in Fig. 5-8. While the

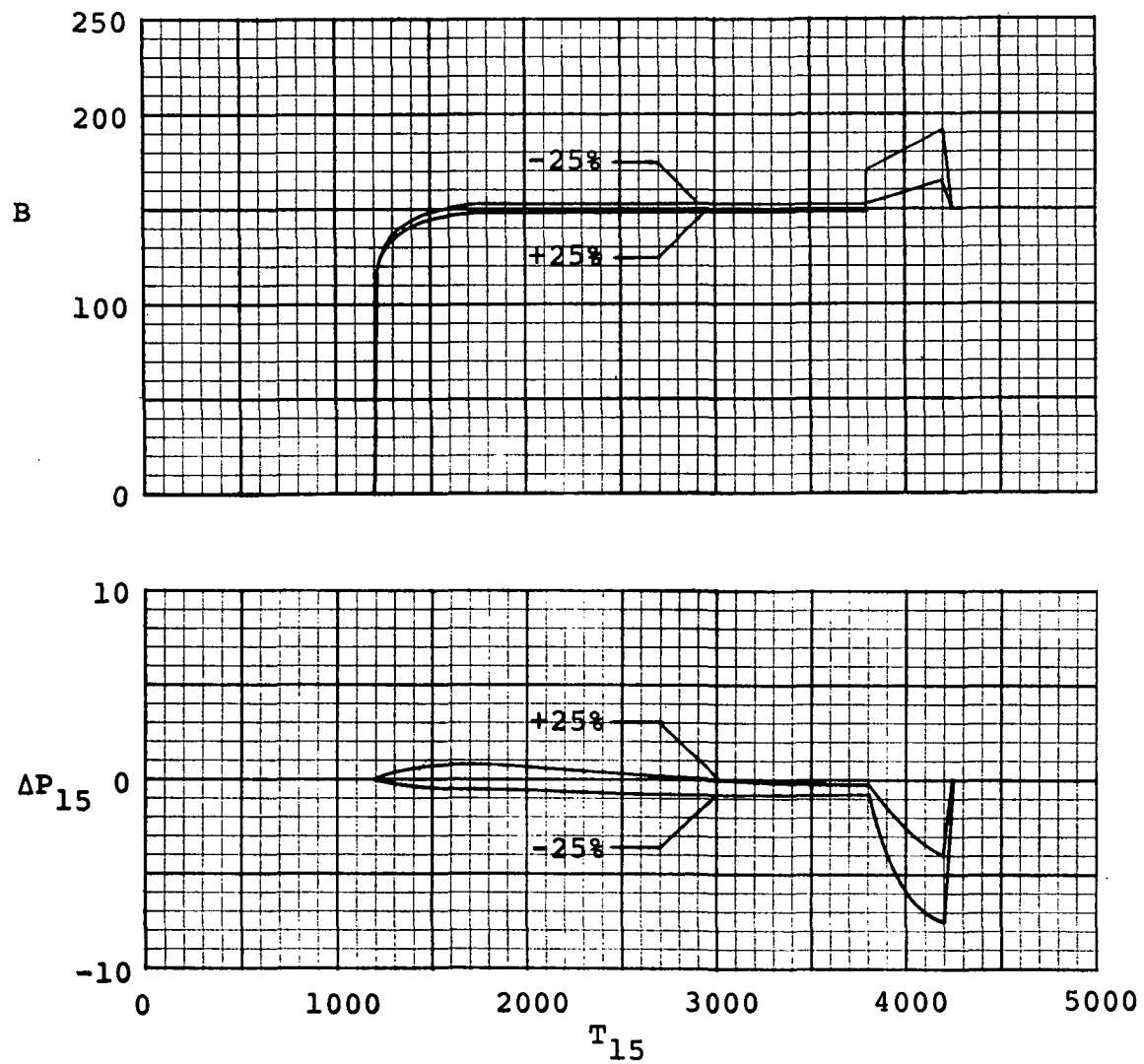


Fig. 5-7. SNM Control System Startup B Trajectory and P_{15} Trajectory Error due to Changes in Turbopump Equation Coefficients

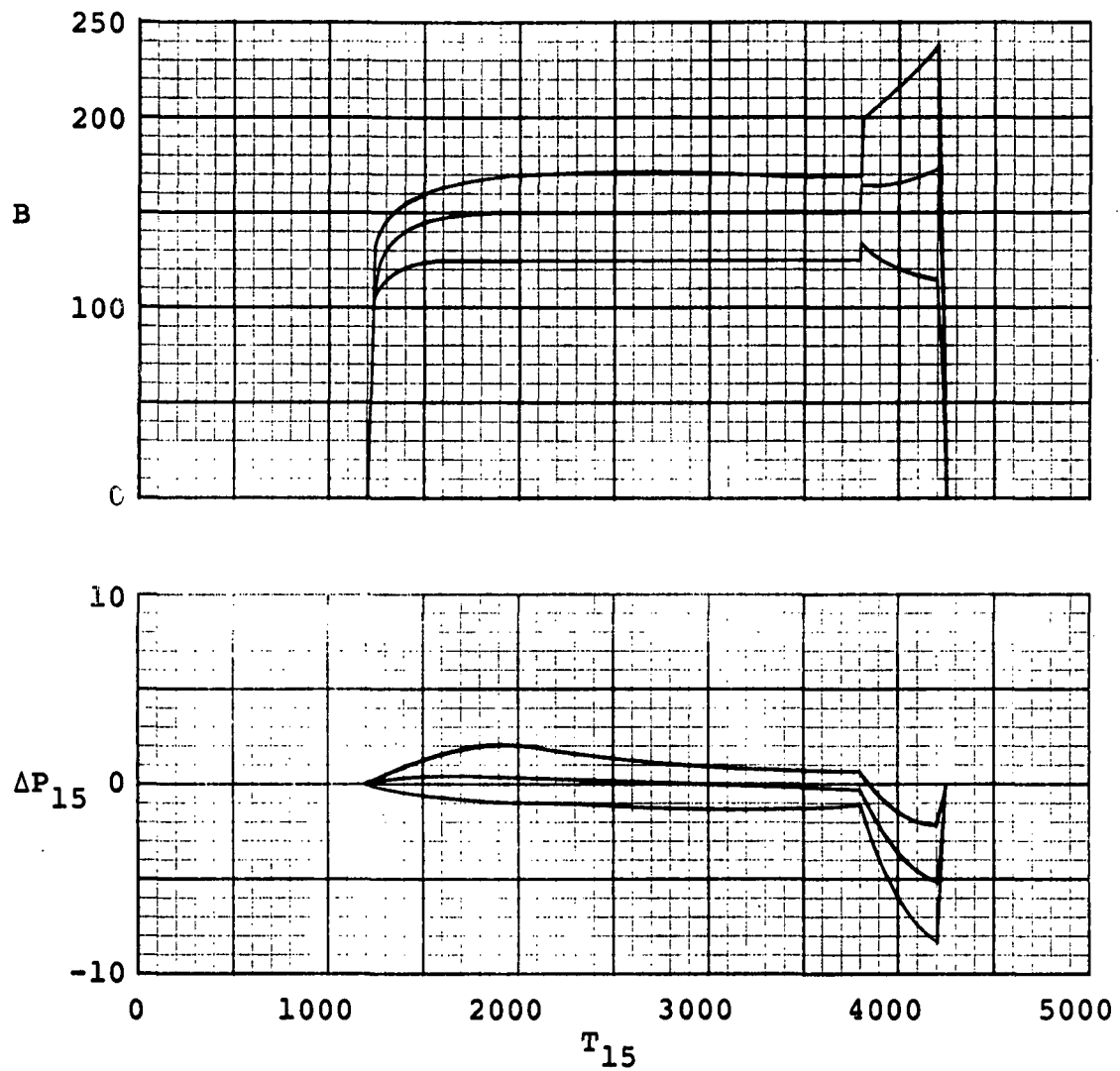


Fig. 5-8. SNM Control System Startup B Trajectory and P_{15} Trajectory Error due to Changes in Temperature Equation Coefficients

resulting P_{15} errors are again minor, the trajectory in the (T_{15}, B) plane is more strongly affected. Greater sensitivity to temperature parameter variations is to be expected since B is so intimately connected with these parameters. System performance is probably acceptable for these parameter variations, but variations larger than ± 25 percent should be avoided. This is not unreasonable; it should not be difficult to determine these model coefficients to within ± 25 percent.

Summary

Operating characteristics of the CM and SNM control systems are presented in this chapter. The CM control system is an application of the PSDC method in which an exact model of the plant is available and in which the control vector is a function of the entire plant state vector. The CM control system exhibits excellent trajectory following properties, an almost unlimited state space region wherein recovery to the nominal trajectory is possible, and is almost insensitive to certain system parameter variations.

The SNM controller design is based on many modeling approximations, and the control vector is a function of a small subset of the plant state vector. Nevertheless the SNM control system follows the specified startup/shutdown trajectory closely. The system is capable of returning to the nominal trajectory from initial conditions within a

strip 150 psi wide, centered on the nominal trajectory, in the (T_{15}, P_{15}) plane. Finally, the system is relatively insensitive to turbopump time constant variations and reasonably tolerant to variations in heat transfer time constants.

CHAPTER 6

SUMMARY AND CONCLUSIONS

Summary

This study deals with control of the NERVA nuclear rocket engine. The most detailed model of NERVA is one developed by the Aerojet and Westinghouse Corporations, and is called the Common Analog Model. Because the CAM is too complex to provide a basis for control system design, Kendrick (1972) has developed a Simplified Nonlinear Model (SNM) which has been shown to retain the important characteristics of the CAM. The SNM is the reference model used in this study.

The main purpose of this study, then, is to design a digital control system for the SNM. This is accomplished by using the Precomputed State Dependent Control method. The method is applicable over a broad class of systems, and involves the synthesis of a control law which approximates an optimum. Optimal system behavior is explicitly defined as a state space trajectory, rather than by some integral performance index, and the control law synthesis is performed on this basis. Precomputed State Dependent Control is most practically implemented by an on-line digital computer.

Application of the PSDC method is described in mathematical terms including the off-line controller design and on-line system control phases. Two preliminary examples of the method are given. The first is a linear system containing three states and two controls. The second involves a highly simplified model of a NERVA class nuclear rocket engine developed by Smith and Stenning (1961).

Because of the relatively high order and basic form of the SNM, the PSDC method is not readily applicable to this model. For this reason a lower order Control Model is derived from the SNM to form the basis for the SNM controller design. A number of approximations are made in this derivation. The result of one approximation involving the engine neutronics, is that a slight modification in trajectory specification is necessary in the off-line SNM controller design.

Three aspects of the SNM control system performance are examined: the ability of the system to follow the nominal trajectory, its ability to return to the nominal from distant initial conditions, and the sensitivity of the controller to variations in the dominant plant time constants. System performance is certainly acceptable in all three aspects. Examination of the on-line controller operation shows that these desirable control characteristics are achieved using very few on-line state measurements, and that the required computer capability is minimal.

The main contributions of this study are the development of the Control Model and the introduction of the PSDC concept. While the Control Model may find little application outside this work, its validity is demonstrated by the performance of the SNM control system. The PSDC method itself is, of course, more generally applicable.

Conclusions

Based on the experience gained here it can be concluded that the PSDC method is generally applicable over a class of control problems in which the desired system performance can be expressed as a trajectory in state space. Advantages of the PSDC method are listed below.

1. The method is applicable to multiple-input multiple-output nonlinear time-invariant systems.
2. Because the control law synthesis is performed off-line, the on-line implementation is conceptually simple and digital computer speed and storage requirements are minimal.
3. State constraints can be implemented in a direct fashion in the selection of a nominal trajectory.
4. No reference input time functions are required; every aspect of desired system behavior is embedded in the controller itself.

The advantages themselves suggest certain limitations. Some are common to many control schemes and some are peculiar to PSDC. These are as follows.

1. The method requires an explicit specification of a nominal trajectory which cannot be modified without re-doing the off-line control law synthesis. (See recommendations.)

2. The nature of the method is such that control must be implemented digitally either by a small special purpose computer or by time sharing a larger computer. This limits the application of the method to cases where such expense is justified.

3. Control constraints are difficult to implement directly.

4. A fairly accurate model of the plant to be controlled is required.

The requirement that the desired system performance be expressed as a trajectory in state space could be viewed as a limitation. However, in many instances this is the best means of expression. Also, behavior which is normally expressed as a time response can be expressed equivalently as a trajectory in some cases. In regard to number 3 above, all schemes which involve control law synthesis suffer the same difficulty in implementing control constraints directly. It should be pointed out that a great deal of success has been achieved in this work by implementing these

constraints a posteriori. Though system behavior under the effects of these constraints cannot be described as "optimal" in any particular sense, in every case examined, the system quickly recovered to conditions which no longer called for the violation of these constraints. Hence, such constraints need not rule out the applicability of PSDC.

Regarding the specific PSDC application in the SNM control system design, several other conclusions are reached. First, the Control Model is seen to be a good basis for the SNM controller design, as verified by the SNM control system performance. The latter exhibits excellent trajectory following properties and is reasonably insensitive to changes in the dominant SNM time constants. In comparison with other methods proposed for the control of NERVA, this method requires fewer on-line measurements and appears to have a much smaller on-line digital computer requirement, both in speed and storage capability.

Suggestions for Further Study

The success achieved thus far in the application of the PSDC method to the SNM control problem indicates some further study would be worthwhile. Suggestions for additional effort in three areas are made below.

1. As pointed out in Chapter 4, the hyperellipsoid control law form used throughout applicable state space contains some terms which are relatively unimportant. It

should be possible to reduce the number of coefficients in each control law by refitting the optimal control data to a form which excludes these terms, without degrading the laws significantly.

2. The basic PSDC method requires that a nominal trajectory be specified prior to the off-line synthesis phase of controller design, and hence that the specification not be modified without a complete redesign. It may be possible to simply modify existing laws rather than computing new ones, to correspond to a trajectory specification somewhat different than the original. This would be done by synthesizing a set of control laws for, say, a family of parallel trajectory specifications. Thus the control law coefficients would be functions of the trajectory specification and could be modified at any time.

3. The control system could be made even more immune to variations in plant parameters by making the control law coefficients functions of the most dominant parameters, rather than constants. This would be accomplished in a manner similar to that suggested in 2 above. Another means of reducing controller sensitivity would be to include external compensation to remove steady-state errors.

APPENDIX A

TEMPERATURE FUNCTION APPROXIMATIONS AND VARIABLE TRANSFORMATION IN THE CONTROL MODEL DEVELOPMENT

Chapter 3 describes the development of the Control Model omitting the details of the temperature function approximations and the final transformation of variables. These aspects of the CM development are given in this appendix.

Temperature Function Approximations

Equations (3-1) through (3-3) are of the form

$$\dot{T}_{15} = a_1(T_{c2} - T_{15}) + b_1 u \quad (A-1)$$

$$\dot{T}_{c2} = a_2(T_{c1} - T_{c2}) + b_2 u \quad (A-2)$$

$$\dot{T}_{c1} = -a_3 T_{c1} + b_3 u \quad (A-3)$$

Assume zero initial conditions and take the Laplace transform of the above.

$$(s + a_1)T_{15} - a_2 T_{c2} = b_1 U \quad (A-4)$$

$$(s + a_2)T_{c2} - a_2 T_{c1} = b_2 U \quad (A-5)$$

$$(s + a_3)T_{c1} = b_3 U \quad (A-6)$$

Suppose $u(t) = kt$, or $U(s) = k/s^2$. Then (A-6) can be solved

for $T_{c1}(s)$ giving

$$T_{c1}(s) = \frac{b_3 k / a_3}{s^2} - \frac{b_3 k / a_3^2}{s} + \frac{b_3 k / a_3^2}{s + a_3} \quad (A-7)$$

$$T_{c1}(t) = \frac{b_3 k}{a_3} \left(t - \frac{1}{a_3} + \frac{1}{a_3} e^{-a_3 t} \right) \quad (A-8)$$

The third term in (A-8) becomes negligible for large t . If this transient is ignored, the corresponding term in (A-7) is dropped.

$$T_{c1}(s) \approx \frac{k \left(\frac{b_3}{a_3} \right)}{s^2} - \frac{k \left[\frac{b_3}{a_3} \left(\frac{1}{a_3} \right) \right]}{s} \quad (A-9)$$

Now (A-2) can be solved for $T_{c2}(s)$ using (A-9). The transient term in this solution is also dropped leaving

$$T_{c2}(s) \approx \frac{k \left(\frac{b_2}{a_2} + \frac{b_3}{a_3} \right)}{s^2} - \frac{k \left[\frac{b_3}{a_3} \left(\frac{1}{a_2} + \frac{1}{a_3} \right) + \frac{b_2}{a_2} \left(\frac{1}{a_2} \right) \right]}{s} \quad (A-10)$$

Similarly, (A-1) is solved for $T_{15}(s)$ and its transient term dropped, leaving

$$T_{15}(s) \approx \frac{k(\frac{b_1}{a_1} + \frac{b_2}{a_2} + \frac{b_3}{a_3})}{s^2} - \frac{k \left[\frac{b_3}{a_3}(\frac{1}{a_1} + \frac{1}{a_2} + \frac{1}{a_3}) + \frac{b_2}{a_2}(\frac{1}{a_1} + \frac{1}{a_2}) + \frac{b_1}{a_1}(\frac{1}{a_1}) \right]}{s}$$

(A-11)

Referring back to equations (3-1), (3-2), and (3-3),

$$\begin{aligned} a_1 &= .02019W_n & b_1 &= 16.25 \\ a_2 &= .02671W_n & b_2 &= 31.28 \\ a_3 &= .01105W_n & b_3 &= 21.31 \end{aligned}$$

(A-12)

Substituting the values from (A-12) into equation (A-11),

$$T_{15}(s) \approx \frac{3904.5k/W_n}{s^2} - \frac{483960k/W_n^2}{s}$$

(A-13)

For large t ,

$$\dot{T}_{15}(s) \approx \frac{3904.5k/W_n}{s}$$

(A-14)

Now suppose that $T_{c1}(s)$ can be expressed as a linear combination of $T_{15}(s)$ and $\dot{T}_{15}(s)$.

$$T_{c1}(s) = \gamma T_{15}(s) + \beta \dot{T}_{15}(s)$$

(A-15)

Also, from (A-12),

$$T_{c1}(s) \approx \frac{1928.5k/W_n}{s^2} - \frac{174530k/W_n^2}{s} \quad (A-16)$$

Combine (A-13) through (A-16) and equate coefficients to obtain $\gamma = .4131$ and $\beta = 16.52/W_n$. Finally, after a transformation back into the time domain,

$$T_{c1}(t) \approx .4939T_{15}(t) + \frac{16.52}{W_n} \dot{T}_{15}(t) \quad (A-17)$$

which is the same as equation (3-31) in the CM set.

The approximate expressions for T_7 and T_{11} are found in a similar manner, but first it is necessary to define $T'_7 = T_7 - 45$ and $T'_{11} = T_{11} - 45$. Then equations (3-21) and (3-22) are of the form,

$$\dot{T}'_7 = -a_4 T'_7 + b_4 u \quad (A-18)$$

$$\dot{T}'_{11} = -a_5 T'_{11} + b_5 u \quad (A-19)$$

where

$$\begin{aligned} a_4 &= .011W_{11} & b_4 &= 2.53 \\ a_5 &= .011W_{11} & b_5 &= 1.59 \end{aligned} \quad (A-20)$$

Using the same procedure as for T_{c1} , T_7 and T_{11} are found to be

$$T'_7(t) \approx .03702T_{15}(t) + \frac{1.223}{W_n} \dot{T}_{15}(t) \quad (A-21)$$

$$T'_{11}(t) \approx .05891T_{15}(t) + \frac{1.946}{W_n} \dot{T}_{15}(t) \quad (A-22)$$

and (A-21) and (A-22) are equivalent to (3-32) and (3-33). To arrive at the above it is necessary to assume $W_{11} \approx W_n$, which is a good approximation for this purpose.

Variable Transformation

After the neutronics and temperature approximations are incorporated, the model consists of the following set of equations.

$$\dot{T}_{15} = .02019W_n(T_{c2} - T_{15}) + 16.25S \quad (A-23)$$

$$\dot{T}_{c2} = .02671W_n(T_{c1} - T_{c2}) + 31.28S \quad (A-24)$$

$$T_{c1} = .4939T_{15} + \frac{16.25B}{W_n} \quad (A-25)$$

$$T_7 = .03702T_{15} + \frac{1.223B}{W_n} + 45 \quad (A-26)$$

$$T_{11} = .05891T_{15} + \frac{1.946B}{W_n} + 45 \quad (A-27)$$

$$P_{15} = .076W_n\sqrt{T_{15}} \quad (A-28)$$

$$W_n = W_{bcv} + W_T \quad (A-29)$$

$$W_{bcv} = .04\theta_{bcv} \sqrt{\frac{P_{11}}{T_{11}}(P_{11} - P_{13})} \quad (A-30)$$

$$P_{13} = P_{15} \left(1 + \frac{33.3}{\sqrt{T_{15}}} \right) \quad (A-31)$$

$$\dot{W}_T = 10(W_{11} - W_{bcv} - W_T) \quad (A-32)$$

$$W_{11} = W_{no} + W_{ss} \quad (A-33)$$

$$W_{no} = 1.5 \sqrt{\frac{P_2 + P_{11}}{2T_7}} (P_2 - P_{11}) \quad (A-34)$$

$$W_{ss} = .0184\theta_{ssv} \sqrt{\frac{P_2 + P_{11}}{2T_7}} (P_2 - P_{11}) \quad (A-35)$$

$$P_{11} = .346W_T \sqrt{T_{11}} + .786P_{13} \quad (A-36)$$

$$P_2 = 3.434 \times 10^{-6} N^2 - 2.967 \times 10^{-4} W_{11} N + 30 \quad (A-37)$$

$$\begin{aligned} \dot{N} = & 9.6 \times 10^5 \frac{W_T T_{11}}{N} \left[1 - \left(\frac{P_{13}}{P_{11}} \right)^{.25} \right] - 8.46 \times 10^{-5} N^2 \\ & - .021W_{11} N \end{aligned} \quad (A-38)$$

Equations (A-23) through (A-38) contain four differential equations in which the states are T_{15} , T_{c2} , W_T , and N . The object is to transform this equation set to obtain differential equations in T_{15} , $\dot{T}_{15}$, P_{15} , and N . Also the implicit nature of the algebraic equations must be removed so that simultaneous solutions of nonlinear equations can be avoided in any digital simulation of the CM.

The transformation begins by creating the state $\dot{T}_{15}$ and eliminating the state T_{c2} . With

$$B = \dot{T}_{15} \quad (A-39)$$

by definition, the following are obtained from (A-22).

$$T_{c2} = \frac{B - 16.25S}{.02019W_n} + T_{15} \quad (A-40)$$

$$\dot{B} = .02019[W_n(\dot{T}_{c2} - B) + \dot{W}_n(T_{c2} - T_{15})] \quad (A-41)$$

Note that $\dot{S}$ is assumed to be zero since it is an input and is piecewise constant. Substitute (A-40) into (A-24).

$$\dot{T}_{c2} = .02671W_n(T_{c1} - T_{15}) - 1.323B + 52.78S \quad (A-42)$$

Now substitute (A-40) and (A-42) into (A-41).

$$\begin{aligned} \dot{B} = & .02019W_n[.02671W_n(T_{c1} - T_{15}) - 2.323B + 52.78S] \\ & + \dot{W}_n \frac{B - 16.25S}{W_n} \end{aligned} \quad (A-43)$$

The unknown quantity is $\dot{W}_n$. From (A-20),

$$W_T = W_n - W_{bcv} \quad (A-44)$$

$$\dot{W}_n = \dot{W}_{bcv} + \dot{W}_T \quad (A-45)$$

Now $\dot{W}_{bcv}$ is obtained by taking a derivative of (A-30).

$$\begin{aligned} \dot{W}_{bcv} = & \frac{.0008\theta_{bcv}^2}{T_{11}W_{bcv}} \left[\dot{P}_{11}(2P_{11} - P_{13}) - \dot{P}_{13}P_{11} \right. \\ & \left. - \frac{\dot{T}_{11}P_{11}}{T_{11}}(P_{11} - P_{13}) \right] \end{aligned} \quad (A-46)$$

From (A-37),

$$\dot{P}_{11} = .346 \left(\frac{W_T \dot{T}_{11}}{2\sqrt{T_{11}}} + \dot{W}_T \sqrt{T_{11}} \right) + .786\dot{P}_{13} \quad (A-47)$$

and from (A-32),

$$\dot{P}_{13} = .076 \left[\frac{W_n^B}{2\sqrt{T_{15}}} + \dot{W}_n(\sqrt{T_{15}} + 33.3) \right] \quad (A-48)$$

Substitute (A-46), (A-47), and (A-48) back into (A-45) to obtain,

$$\begin{aligned} \dot{W}_n \left[1 - \frac{6.08 \times 10^{-5} \theta_{bcv}^2 (.572P_{11} - .786P_{13})(T_{15} + 33.3)}{T_{11} W_{bcv}} \right] = \\ \frac{.0008 \theta_{bcv}^2}{T_{11} W_{bcv}} \left\{ .346 \dot{W}_T T_{11} (2P_{11} - P_{13}) + .038 (.572P_{11} \right. \\ \left. - .786P_{13}) \frac{W_n^B}{T_{15}} + \dot{T}_{11} \left[.173 \frac{W_T}{T_{11}} (2P_{11} - P_{13}) \right. \right. \\ \left. \left. - \frac{P_{11}}{T_{11}} (P_{11} - P_{13}) \right] \right\} + \dot{W}_T \quad (A-49) \end{aligned}$$

where $\dot{W}_T$ is given by (A-31) and $\dot{T}_{11}$ by the original SNM equation (3-22). Thus (A-49) can be solved for $\dot{W}_n$ and written in the form,

$$\dot{W}_n = f_3(W_{bcv}, W_T, W_n, W_{11}, P_{11}, P_{13}, T_{11}, T_{15}, B, \theta_{bcv}) \quad (A-50)$$

The substitution of (A-50) back into (A-43) completes the derivation of $\dot{B}$.

A differential equation involving P_{15} is obtained by differentiating (A-28),

$$\dot{P}_{15} = .076 \left[\frac{W_n B}{2\sqrt{T_{15}}} + \dot{W}_n \sqrt{T_{15}} \right] \quad (A-51)$$

and substituting (A-50) into (A-51). Thus the states P_{15} and B have been created and states T_{c2} and W_T have been "destroyed".

The four state (differential) equations (A-39), (A-43), (A-51), and (A-38) are written in terms of the interim quantities T_7 , T_{11} , W_n , W_{bcv} , P_{13} , W_T , W_{11} , P_{11} , and P_2 , as well as the four states themselves and the two controls S and θ_{bcv} . Some of these interim quantities are explicit functions of state variables, but others form implicit groups which must be solved simultaneously. To avoid the special numerical techniques necessary to integrate systems of equations containing implicit algebraic equations, these implicit "loops" must be removed.

The first implicit loop consists of equations (A-29), (A-30), and (A-36) which involve W_T , W_{bcv} , and P_{11} . Begin by squaring and rearranging (A-30),

$$\frac{W_{bcv}^2 T_{11}}{(.04\theta_{bcv})^2} = P_{11}(P_{11} - P_{13}) \quad (A-52)$$

and substituting (A-36).

$$\frac{W_{bcv}^2 T_{11}}{(.04\theta_{bcv})^2} = (.346W_T T_{11} + .786P_{13}) \times (.346W_T T_{11} - .214P_{13}) \quad (A-53)$$

Substitute (A-20) into (A-53) and collect terms.

$$\begin{aligned} W_{bcv}^2 [T_{11}(625/\theta_{bcv}^2 - .119716)] \\ + W_{bcv} [.692 T_{11}(d_1 + .286P_{13})] \\ + [(.786P_{13} + d_1)(.214P_{13} - d_1)] = 0 \end{aligned} \quad (A-54)$$

where $d_1 = .346\sqrt{T_{11}}W_n$. Equation (A-54) is a quadratic in W_{bcv} and has two solutions. The positive solution is the only physically meaningful one, and is expressed as

$$W_{bcv} = f_1(W_n, T_{11}, P_{13}, \theta_{bcv}) \quad (A-55)$$

Now W_T and P_{11} can be found using equations (A-29) and (A-36), and the implicit loop is broken.

The other implicit loop consists of equations (A-33), (A-34), (A-35), and (A-37) involving W_{11} , W_{no} , W_{ss} , and P_2 . Begin by combining the first three equations.

$$W_{11} = (1.5 + .0184\theta_{ssv}) \sqrt{\frac{P_2 + P_{11}}{2T_7}(P_2 - P_{11})} \quad (A-56)$$

Square and rearrange (A-56).

$$\frac{2W_{11}^2 T_7}{(1.5 + .0184\theta_{ssv})^2} = P_2^2 - P_{11}^2 \quad (A-57)$$

Substitute (A-37) into (A-57) and collect terms.

$$\begin{aligned} W_{11}^2 [d_2 T_7 - .025641 d_3] + W_{11} [.0005934 N (d_3 + 30)] \\ + [P_{11}^2 - d_3 (d_3 + 60) - 900] = 0 \end{aligned} \quad (A-5)$$

$$\text{where } d_2 = \frac{2}{(1.5 + .0184\theta_{ssv})^2} \quad (A-59)$$

$$d_3 = 3.434 \times 10^{-6} N^2 \quad (A-60)$$

Equation (A-59) is quadratic in W_{11} and, as before, only the positive solution is physically meaningful. Thus, W_{11} can be expressed as

$$W_{11} = f_2(P_{11}, N, T_7) \quad (A-61)$$

and the last implicit loop is broken. Since P_2 does not appear in any of the CM state equations, (A-37) is excluded from the final CM equation set.

To conclude, the CM equation set includes the inter-im variables P_{13} , W_n , T_{cl} , T_7 , T_{11} , W_{bcv} , W_T , P_{11} , W_{11} , and W_{nd} (or $\dot{W}_n$). No simultaneous algebraic solutions are required when these quantities are calculated in the order given by equations (3-29) through (3-38).

APPENDIX B

COMPUTER PROGRAMS

This appendix contains listings of two FORTRAN programs which are representative of the off and on-line computations performed in designing and implementing a PSDC controller. The first is a program which performs a series of optimizations within one or more blocks of state space to obtain sets of optimal control vectors as described in Chapter 2. Provisions are made to read in information regarding the nominal trajectory specification, block geometry, convergence parameters, etc. The second program is one which performs a digital simulation of the SNM control system in the startup mode, as used to obtain part of the data presented in Chapter 5.

Off-Line Optimization Program

```

PROGRAM MAIN(INPUT,OUTPUT,PUNCH,TAPE 1=INPUT)
COMMON /B1/ DTI,DTI2,SK1(29)
COMMON /B2/ SK2(9),U0(2),U01(25),U02(25)
COMMON /B3/ DT,EPS,DU,TN(4)
COMMON /B4/ SK4(4),M,SK3
COMMON /B5/ SK5(4),XC(25,4),SK6(475)
COMMON SK8(600)
1  READ 100,NB
   IF(E0F,1) 99,2
2  READ 101,EPS,DU,DT,DTI,TN,((XC(I,J),J=1,4),I=1,25)
   READ 101,U01,U02
   M=DT/DTI+.5
   DTI=DT/M
   DTI2=DTI/2.
   PRINT 200,NB
   PRINT 201,EPS,DU,DT,DTI,TN
   CALL FITMAT
   CALL BLOCK
   GO TO 1
99 STOP
100 FORMAT(I3)
101 FORMAT(4E20.13)
200 FORMAT(*1BLOCK*I4/)
201 FORMAT(* EPS=*E19.8/* DU=*E20.8/* DT=*E20.8/* DTI=*
C E19.8/* TN=*4E20.8)
END

```

```

SUBROUTINE BLOCK
COMMON /B1/ SK5(6),DT0(25)
COMMON /B2/ I,IC,SK2(4),UN(2),SK3,U0(2),U01(25),
C U02(25)
COMMON /B4/ X(4)
COMMON /B5/ SK1(4),XC(25,4),SK4(375),XA(25,4)
COMMON UC1(15),UC2(15),U1(25),U2(25)
DO 80 I=1,25
DO 78 K=1,4
78 X(K)=XC(I,K)
   UN(1)=U01(I)
   UN(2)=U02(I)
80 CALL OPT(U1(I),U2(I))
90 PRINT 305,U1,U2
   CALL FIT(U1,UC1)
   CALL CHECK(U1,UC1)

```

```

CALL FIT(U2,UC2)
CALL CHECK(U2,UC2)
PRINT 302,UC1,UC2
RETURN
302  FORMAT(* UC1=*/3(5E20.8/)/ * UC2=*/3(5E20.8/)/)
305  FORMAT(/ * U1=*/5(5E20.8/)* U2=*/5(5E20.8/))
306  FORMAT(4E20.13)
END

```

```

SUBROUTINE OPT(UA,UB)
COMMON /B1/ DT1,DTI2,SK2(4),DT0(25)
COMMON /B2/ I,IC,B,YR,G(2),UN(2),IG
COMMON /B3/ DT,SK1(2),TN(4)
COMMON /B4/ X(4),M,CX
DIMENSION P1(2),P3(2)
K=0
B=.1
IC=0
IG=0
CALL STAGE(X,CX)
CALL COST(UN,YR)
P1(1)=UN(1)
P1(2)=UN(2)
CALL GRAD
IF(IG.EQ.1) GO TO 15
CALL SEARCH(1)
P3(1)=UN(1)
P3(2)=UN(2)
3   IF(K.GT.19) GO TO 20
K=K+1
CALL GRAD
IF(IG.EQ.1) GO TO 15
CALL SEARCH(1)
IF(IC.EQ.1) GO TO 16
G(1)=(UN(1)-P1(1))/3.
G(2)=(UN(2)-P1(2))/3.
CALL SEARCH(0)
DO 7 J=1,2
P1(J)=P3(J)
7   P3(J)=UN(J)
GO TO 3
15  PRINT 207,I
GO TO 16
20  PRINT 206,I
16  UA=UN(1)
UB=UN(2)

```

```

PRINT 301,I,UA,UB,YR,DT,K,M
RETURN
301 FORMAT(* POINT*I4* U1,U2,COST,DT,K,M=*4E20.8,2I7)
206 FORMAT(* POINT*I4* DID NOT CONVERGE*)
207 FORMAT(* ZERO GRADIENT AT POINT*I4)
END

```

```

SUBROUTINE SEARCH(KK)
COMMON /B2/ SK1,IC,B,YR,G(2),UN(2)
COMMON /B3/ SK2,EPS
DIMENSION U(2)
K=0
Y1=YR
A=1.
U(1)=UN(1)+G(1)
U(2)=UN(2)+G(2)
CALL COST(U,YA)
IF(YA.LE.Y1) GO TO 9
4 Y3=YA
A3=A
A=A/3.
U(1)=UN(1)+A*G(1)
U(2)=UN(2)+A*G(2)
CALL COST(U,YA)
IF(YA.LE.Y1) GO TO 12
IF(KK.EQ.0) GO TO 14
B=B/3.
IF(B.LE.EPS) GO TO 42
GO TO 4
14 K=K+1
IF(K.GE.2) RETURN
GO TO 4
12 Y2=YA
A2=A
A1=0.
GO TO 20
9 SS=1.
10 Y2=YA
SS=2.*SS
A=A+SS
U(1)=UN(1)+A*G(1)
U(2)=UN(2)+A*G(2)
CALL COST(U,YA)
IF(KK.NE.0) B=2.*B
IF(YA.LE.Y2) GO TO 18
Y3=YA

```

```

A3=A
A2=A-SS
A1=A2-SS/2.
20 DA=Y1*(A3-A2)
DB=Y2*(A1-A3)
DC=Y3*(A2-A1)
A3=(DA*(A3+A2)+DB*(A1+A3)+DC*(A2+A1))*0.5/(DA+DB+DC)
UN(1)=UN(1)+A3*G(1)
UN(2)=UN(2)+A3*G(2)
CALL COST(UN,YR)
RETURN
18 Y1=Y2
GO TO 10
42 IC=1
RETURN
END

```

```

SUBROUTINE STAGE(X,C)
COMMON /B3/ SK1(3),TN(4)
DIMENSION X(4)
DA=X(2)-TN(1)*X(1)-TN(2)
DB=X(3)-TN(3)*X(1)-TN(4)
C=SQRT(DA**2+DB**2)
RETURN
END

```

```

SUBROUTINE COST(U,Y)
COMMON /B1/ DTI,DTI2,XN(4)
COMMON /B4/ X(4),M,CX
DIMENSION U(2)
DO 3 I=1,4
3 XN(I)=X(I)
DO 7 I=1,M
7 CALL RUNKUT(U)
CALL STAGE(XN,Y)
RETURN
END

```

```

SUBROUTINE GRAD
COMMON /B2/ SK1(2),B,YR,G(2),UN(2),IG
COMMON /B3/ SK2(2),DU
DIMENSION U(2)
U(1)=UN(1)-DU
U(2)=UN(2)
CALL COST(U,Y1)
U(1)=UN(1)
U(2)=UN(2)-DU
CALL COST(U,Y2)
G(1)=Y1-YR
G(2)=Y2-YR
GM=AMAX1(ABS(G(1)),ABS(G(2)))
IF(GM.EQ.0.) GO TO 4
G(1)=B*G(1)/GM
G(2)=B*G(2)/GM
RETURN
4 IG=1
RETURN
END

```

```

SUBROUTINE RUNKUT(U)
COMMON /B1/ DT1,DTI2,XN(4)
COMMON /B5/ F(4)
DIMENSION U(2),C1(4),C2(4),C3(4),XA(4)
CALL FUN(XN,U)
DO 2 I=1,4
C1(I)=DTI2*F(I)
2 XA(I)=XN(I)+C1(I)
CALL FUN(XA,U)
DO 3 I=1,4
C2(I)=DTI2*F(I)
3 XA(I)=XN(I)+C2(I)
CALL FUN(XA,U)
DO 4 I=1,4
C3(I)=DTI*F(I)
4 XA(I)=XN(I)+C3(I)
CALL FUN(XA,U)
DO 5 I=1,4
5 XN(I)=XN(I)+(2.*(C1(I)+C3(I))+4.*C2(I)+DTI*F(I))/6.
RETURN
END

```

```

SUBROUTINE FUN(X,U)
COMMON /B5/ F(4)
DIMENSION X(4),U(2)
SRX1=SQRT(X(1))
P13=X(3)*(1.+33.3/SRX1)
WN=X(3)/.076/SRX1
DA=X(2)/WN
TC1=.4939*X(1)+16.52*DA
T11=.05891*X(1)+1.946*DA+45.
T7=.03702*X(1)+1.223*DA+45.
SRT11=SQRT(T11)
DA=.346*WN*SRT11
A=T11*(1250./U(2)**2-.239432)
B=.692*SRT11*(DA+.286*P13)
DB=.786*P13
C=(DB+DA)*(.214*P13-DA)
WBCV=(-B+SQRT(B**2-2.*A*C))/A
WT=WN-WBCV
P11=.346*WT*SRT11+DB
DC=3.434E-6*X(4)**2
A=1.3196*T7-.05127*DC
B=5.9340E-4*X(4)*(DC+30.)
C=P11**2-DC*(DC+60.)-900.
W11=(-B+SQRT(B**2-2.*A*C))/A
WTD=(W11-WN)*10.
T11D=-.011*W11*(T11-45.)+2.53*U(1)
DA=.0008*U(2)**2/T11/WBCV
DC=.572*P11-DB
B=1.-DA*.076*DC*(SRX1+33.3)
A=2.*P11-P13
WND=(DA*(.346*WTD*SRT11*A+.038*DC*WN*X(2)/SRX1+T11D*
C (.173*WT/SRT11*A-P11/T11*(P11-P13)))+WTD)/B
F(1)=X(2)
F(2)=.02019*WN*(.02671*WN*(TC1-X(1))-2.323*X(2)+52.78
C *U(1))+WND*(X(2)-16.25*U(1))/WN
F(3)=.076*(WN*X(2)/SRX1/2.+WND*SRX1)
F(4)=9.6E+5*WT*T11/X(4)*(1.-(P13/P11)**.25)-X(4)*
C (8.46E-5*X(4)+.021*W11)
RETURN
END

```

```

SUBROUTINE CHECK (U,UC)
CCCC U=EXACT DATA, UC=FIT COEF. VECTOR.
COMMON /B4/ X(4)
COMMON /B5/ SK1(479),XC(25,4)
DIMENSION U(25),UC(15)
EMAX=0.
ERROR=0.
QA=0.
DO 8 I=1,25
DO 7 J=1,4
7 X(J)=XC(I,J)
CALL QUAD(UC,X,Q)
QA=QA+Q
DU=ABS(Q-U(I))
IF(DU.GT.EMAX) EMAX=DU
8 ERROR=ERROR+DU
AVG=ERROR/25.
QA=QA/25.
PRINT 201,AVG,EMAX,QA
RETURN
201 FORMAT(/* AVERAGE AND MAXIMUM ERROR=*2E20.8,12X
C *AVERAGE VALUE=*E20.8/)
END

```

```

SUBROUTINE QUAD (A,X,Q)
CCCC A=COEFFICIENT VECTOR, X=COORDINATES, Q=RESULT.
DIMENSION A(15),X(4)
Q=(A(8)*X(3)+A(7)+A(9)*X(2)+A(10)*X(1))*X(3)+(A(5)*
C X(2)+A(4)+A(6)*X(1))*X(2)+A(1)+(A(3)*X(1)+A(2))*X(1)
C +(A(12)*X(4)+A(11)+A(13)*X(3)+A(14)*X(2)+A(15)*X(1))
C *X(4)
RETURN
END

```

```

SUBROUTINE FIT(C,A)
COMMON /B5/ SK1(104),FA(15,25)
DIMENSION C(25),A(15)
DO 6 I=1,15
A(I)=0.
DO 6 J=1,25
6 A(I)=A(I)+FA(I,J)*C(J)
RETURN
END

```

```

SUBROUTINE FITMAT
COMMON /B5/ SK2(4),XA(25,4),FA(15,25),XC(25,4)
COMMON F(25,15),FB(15,15)
DO 6 I=1,25
DO 6 J=1,4
6   XC(I,J)=XA(I,J)-XA(2,J)
PRINT 200
DO 13 I=1,25
13  PRINT 201,1,(XA(I,J),J=1,4)
DO 15 K=1,25
F(K,1)=1.
F(K,2)=XC(K,1)
F(K,3)=XC(K,1)**2
F(K,4)=XC(K,2)
F(K,5)=XC(K,2)**2
F(K,6)=XC(K,2)*XC(K,1)
F(K,7)=XC(K,3)
F(K,8)=XC(K,3)**2
F(K,9)=XC(K,3)*XC(K,2)
F(K,10)=XC(K,3)*XC(K,1)
F(K,11)=XC(K,4)
F(K,12)=XC(K,4)**2
F(K,13)=XC(K,4)*XC(K,3)
F(K,14)=XC(K,4)*XC(K,2)
15  F(K,15)=XC(K,4)*XC(K,1)
DO 11 K=1,15
DO 11 J=K,15
FB(J,K)=0.
DO 10 I=1,25
10  FB(J,K)=FB(J,K)+F(I,K)*F(I,J)
11  FB(K,J)=FB(J,K)
CALL MATRIX(10,15,15,0,FB,15,DET,0,0)
PRINT 203,DET
DO 12 K=1,15
DO 12 J=1,25
FA(K,J)=0.
DO 12 I=1,15
12  FA(K,J)=FA(K,J)+FB(K,I)*F(J,I)
RETURN
200  FORMAT(/* COORDINATES OF OPTIMIZATION POINTS*)
201  FORMAT(I4,4E20.8)
203  FORMAT(/* DET=*E20.13/)
END

```

Simplified Nonlinear Model
Digital Simulation Program

```

PROGRAM SIM(INPUT,OUTPUT,TAPE 1=INPUT)
CCCC INDICIES ARE DEFINED AS FOLLOWS. A NEW CONTROL IS
CCCC COMPUTED EVERY N1 STEPS. STATE AND CONTROL IS
CCCC PRINTED EVERY N2 STEPS. N3 TOTAL STEPS ARE COMPUTED.
CCCC DTI IS THE INTEGRATION STEP SIZE, AND X IS THE
CCCC INITIAL STATE. (N1.LE.N2)
COMMON /B2/ F(10),DTI,DTI2,IZ,P15,DKT,S
COMMON /B5/ X(10),U(2)
COMMON UC1(15,8),UC2(15,8),U1(15),U2(15),X0(4,8),
C XR(4),TD(3)
DO 3 J=1,8
READ 101,(X0(I,J),I=1,4)
READ 101,(UC1(I,J),I=1,15),(UC2(I,J),I=1,15)
PRINT 203,J,(X0(I,J),I=1,4)
3 PRINT 204,(UC1(I,J),I=1,15),(UC2(I,J),I=1,15)
1 READ 102,N1,N2,N3,DTI,(X(I),I=1,10)
IF(EOF,1) 99,2
2 READ 101,U,TD(2),TD(3)
T=0.
DUM=DTI*N1
PRINT 201,N1,N2,N3,DTI
I=N1
J=N2
K=0
NO=0
DTI2=DTI/2.
4 K=K+1
TD(1)=TD(2)
TD(2)=TD(3)
TD(3)=X(1)
TDOT=(TD(1)-4.*TD(2)+3.*TD(3))/DTI/2.
IF(I.LT.N1) GO TO 10
N=(X(1)-680.)/519.9999
IF(N.LT.1) N=1
IF(N.LT.6) GO TO 6
N=(X(1)-3600.)/199.9999+5.
IF(N.GT.8) N=8
6 IF(N.EQ.NO) GO TO 8
DO 7 M=1,15
U1(M)=UC1(M,N)
7 U2(M)=UC2(M,N)
NO=N
8 IZ=0
CALL FUN(X,U)
IZ=1
XR(1)=X(1)-X0(1,N)
XR(2)=TDOT-X0(2,N)

```

```

XR(3)=P15-X0(3,N)
XR(4)=X(8)-X0(4,N)
CALL QUAD(U2,XR,U(2))
CALL QUAD(U1,XR,SD)
DU=(1.-DKT)*(1.-S/SD)
IF(U(1).GT.0.) U(1)=-0.0001
DA=SQRT((U(1)+4.5)/4.5)
DR=.471239*SIN(2.*ASIN(DA))*DUM
UD=U(1)+DU
U(1)=UD
IF(ABS(DR).LT.ABS(DU)) U(1)=U(1)-DU+SIGN(DR,DU)
IF(U(2).GT.100.) U(2)=100.
IF(U(2).LT.0.) U(2)=.001
I=1
GO TO 12
10 I=I+1
12 IF(J.LT.N2) GO TO 14
PRINT 202,T,X(1),TDOT,P15,X(8),SD,UD,U(1),U(2),N
J=1
GO TO 16
14 J=J+1
16 T=T+DTI
CALL RUNKUT
IF(K.GT.N3) GO TO 1
GO TO 4
99 STOP
101 FORMAT(4E20.13)
102 FORMAT(3I3,E11.3/(4E20.13))
201 FORMAT(*1KENDRICK MODEL SIMULATION*10X*N1,N2,N3,DTI=*
C 3I6,E20.4//* TIME*9X*T15*10X*T15D*11X*P15*11X*N*12X
C *SD*12X*UD*11X*U(1)*10X*U(2)*7X*NB*)
202 FORMAT(F7.3,8E14.5,I5)
203 FORMAT(//* BLOCK*I4,10X,4E20.8/)
204 FORMAT(3(5E20.8/)/3(5E20.8/))
END

```

```

SUBROUTINE QUAD (A,X,Q)
CCCC A=COEFFICIENT VECTOR, X=COORDINATES, Q=RESULT.
      DIMENSION A(15),X(4)
      Q=(A(8)*X(3)+A(7)+A(9)*X(2)+A(10)*X(1))*X(3)+(A(5)*
C X(2)+A(4)+A(6)*X(1))*X(2)+A(1)+(A(3)*X(1)+A(2))*X(1)
C +(A(12)*X(4)+A(11)+A(13)*X(3)+A(14)*X(2)+A(15)*X(1))
C *X(4)
      RETURN
      END

```

```

SUBROUTINE FUN(X,U)
  DIMENSION X(10),U(2)
  COMMON /B2/ F(10),SK1(2),IZ,P15,DKT,S
  SRX1=SQRT(X(1))
  SRT11=SQRT(X(10))
  D1=SRX1+33.3
  D2=.015041312*SRT11*D1
  D3=.000971546304*D1**2
  A=2.*(625.*X(10)/U(2)**2+D3)
  B=X(7)*(2.*D3-D2)
  C=X(7)**2*(D3-D2-.119716*X(10))
  WN=X(7)/(-B+SQRT(B**2-2.*A*C))/A
  DB=.076*WN
  P13=DB*U1
  P11=.346*X(7)*SRT11+.786*P13
  P15=DB*SRX1
  D4=3.434E-6*X(8)**2
  A=1.3196*X(9)-.05127*D4
  B=5.934E-4*X(8)*(D4+30.)
  C=P11**2-D4*(D4+60.)-900.
  W11=(-B+SQRT(B**2-2.*A*C))/A
  P2=D4-2.967E-4*W11*X(8)+30.
  WSS=.13845*W11
  TC=.39741*X(1)+.26*X(2)+.39*X(3)
  DKT=.55*SQRT(WSS)+3.25*(P13+P15)/TC+.0244*(P2+P11)/X(9)
  C=-.00077*(TC-500.)+U(1)
  SP=.0276*X(4)+.235*X(5)+1.65*X(6)
  S=SP/(1.-DKT)/660.
  IF(IZ.EQ.0) RETURN
  F(1)=.02019*WN*(X(2)-X(1))+16.25*S
  F(2)=.02671*WN*(X(3)-X(2))+31.28*S
  F(3)=-.01105*WN*X(3)+21.31*S
  F(4)=181.*S-.0276*X(4)
  F(5)=369.*S-.235*X(5)
  F(6)=110.*S-1.65*X(6)
  F(7)=(W11-WN)*10.
  F(8)=9.6E+5*X(7)*X(10)/X(8)*(1.-(P13/P11)**.25)-X(8)*
  C(8.46E-5*X(8)+.021*W11)
  F(9)=-.011*W11*(X(9)-45.)+1.59*S
  F(10)=-.011*W11*(X(10)-45.)+2.53*S
  RETURN
  END

```

```
SUBROUTINE RUNKUT
COMMON /B2/ F(10),DTI,DTI2
COMMON /B5/ XN(10),U(2)
DIMENSION C1(10),C2(10),C3(10),XA(10)
CALL FUN(XN,U)
DO 2 I=1,10
  C1(I)=DTI2*F(I)
2  XA(I)=XN(I)+C1(I)
  CALL FUN(XA,U)
  DO 3 I=1,10
    C2(I)=DTI2*F(I)
3  XA(I)=XN(I)+C2(I)
  CALL FUN(XA,U)
  DO 4 I=1,10
    C3(I)=DTI*F(I)
4  XA(I)=XN(I)+C3(I)
  CALL FUN(XA,U)
  DO 5 I=1,10
5  XN(I)=XN(I)+(2.*(C1(I)+C3(I))+4.*C2(I)+DTI*F(I))/6.
  RETURN
END
```

REFERENCES

- Bellman, Richard E., and Dreyfus, Stuart E. Applied Dynamic Programming. Princeton, N. J.: Princeton University Press, 1962.
- Breakwell, John V., Speyer, Jason L., and Bryson, Arthur E. Optimization and Control of Nonlinear Systems Using the Second Variation. J.S.I.A.M. Control, Ser. A, Vol. 1, No. 2, 193, 1963.
- Chang, C. S., and DeRusso, P. M. An Approximate Method of Solving Optimal Control Problems. IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, 375, Oct. 1964.
- Crouch, Holmes F. Nuclear Space Propulsion. Granada Hills, Cal.: Astronuclear Press, 1965.
- Eisenberg, Barry R., and Sage, Andrew P. Closed Loop Optimization of Fixed Configuration Systems. Int. Journal of Control, Vol. 3, No. 2, 183, 1966.
- Graham, E. U., and Strauss, J. C. A Simplification Method for Suboptimal Controller Synthesis. JACC, 829, 1970.
- Harkins, Alvin The Use of Parallel Tangents in Optimization; in Chemical Engineering Progress Symposium Series, Vol. 60, No. 50, 35, 1964.
- Johnson, C. D., and Gibson, J. E. Optimal Control with Quadratic Performance Index and Fixed Terminal Time. IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, 355, Oct. 1964.
- Kelley, Henry J. An Optimal Guidance Approximation Theory. IEEE Transactions on Automatic Control, Vol. AC-9, No. 4, 375, Oct. 1964.
- Kendrick, Larry Ellis On-Line Digital Computer Control of the NERVA Nuclear Rocket Engine. Ph.D. Dissertation, University of Arizona, 1972.
- Larson, Robert E. A Survey of Dynamic Programming Computational Procedures. IEEE Transactions on Automatic Control, Vol. AC-12, No. 6, 767, Dec. 1967.

- Larson, Robert E. State Increment Dynamic Programming. New York: American Elsevier Publishing Co., Inc., 1968.
- Levine, Martin D. Parameterized Feedback Control of Non-linear Processes. *Int. Journal of Control*, Vol. 3, No. 2, 39, 1966.
- Longmuir, A. G., and Bohn, Erik V. The Synthesis of Suboptimal Feedback Control Laws. *IEEE Transactions on Automatic Control*, Vol. AC-12, No. 6, 755, Dec. 1967.
- Love, Carl G., and Lavi, Abraham Evaluation of Feedback Structures. *JACC*, 1081, 1968.
- Luh, J. Y. S., and Shafran, J. S. An Approximate Minimal Time Closed-Loop Controller for Processes with Bounded Control Amplitudes and Rates. *IEEE Transactions on Automatic Control*, Vol. AC-15, No. 2, 185, April 1970.
- Mendel, J. M., and Zapalac, J. J. Realization of a Suboptimal Controller by Off-Line Training Techniques. *JACC*, 258, 1967.
- Meyer, Roland On the Use of Piecewise-Linear Functions in Suboptimal Feedback Control Laws. *IEEE Transactions on Automatic Control*, Vol. AC-16, No. 1, 89, Feb. 1971.
- Mohler, Ronald R. Optimal Nuclear Rocket Engine Control; in Neutron Dynamics and Control. U. S. Atomic Energy Comm., 1966.
- Pontryagin, L. S., Boltyanskii, V. G., Grankrelidze, R. V., and Mishchenko, E. F. The Mathematical Theory of Optimal Processes. New York: John Wiley and Sons, Inc., 1962.
- Raju, G. V. S., and Brueggeman, William M. Synthesis of Adaptive Sub-Optimal Feedback Control Laws. *Int. Journal of Control*, Vol. 12, No. 6, 945, 1970.
- Rooy, J. J. A Method of Realizing Quasi-Time Optimal Control by Means of an Approximate Switching Surface. *Int. Journal of Control*, Vol. 11, No. 2, 255, 1970.
- Smith, H. P., Jr. Closed-Loop Dynamics of Nuclear Rocket Engines with Bleed Turbine Drive. *Nuclear Science and Engineering*, Vol. 14, 371, 1962.

Smith, H. P., Jr., and Stenning, A. H. Open Loop Stability and Response of Nuclear Rocket Engines. Nuclear Science and Engineering, Vol. 11, 76, 1961.

Summary Report of Work to Date on the NERVA Digital Control System Design, Report No. NJD25-CSR01-S155. NERVA Integrated Instrumentation and Controls Operation, Aerojet Nuclear Systems Company, Sacramento, Cal. NJD 0025, Sept. 30, 1970.

Wilde, Douglass J., and Beightler, Charles S. Foundations of Optimizations Englewood Cliffs, N. J.: Prentice-Hall Inc., 1967.